

AÑO II N.º 53

**MICROHOBBY**

# AMSTRAD

*Semanal*

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES AMSTRAD

AÑO II N.º 53

**160 Ptas.**

Canarias 165 pts.

**DEPURACION  
DE PROGRAMAS  
RECURSIVOS EN LOGO**

**SOFTWARE**

**MOVIE,**  
REFLEXION  
Y ACCION  
UNIDAS EN MIL  
PANTALLAS

**CONOCE  
EL INTERIOR  
DE TU AMSTRAD**

**PCW SHOW:  
ESTUVIMOS  
EN LONDRES**

**TIRO CON ARCO:  
TRAS LAS HUELLAS  
DE ROBIN HOOD**



JAVIER IGUAL LXXXVI

**HOBBY PRESS**

# AMSTRAD DMP 2000

## NO ENCONTRARA UNA IMPRESORA QUE LE HAGA TAN BUEN PAPEL.



Soportes abatibles que permiten colocar el papel bajo la impresora.



Cómodo sistema de carga frontal del papel.



Admite diferentes anchos de papel, tanto continuo (de 114 a 254 mm.) como hojas sueltas (102 a 241 mm.)

**POR SOLO  
39.500 PTAS + IVA**



- Especialmente recomendada para ordenadores AMSTRAD serie CPC.
- Conectable a cualquier ordenador con interface centronics.
- Velocidad de impresión de 105 caracteres por segundo.

- Gran variedad de tipos de letra, cursiva, alta calidad (NLQ)
- 40, 66, 80 y 132 caracteres por columna.
- Impresión de gráficos punto a punto en diferentes densidades.
- 96 caracteres ASCII y 8 subconjuntos internacionales.

*¡¡ Increíble !!*

**AMSTRAD** ESPAÑA

GRUPO INDESCOMP



# AMSTRAD

## Sumario

**Director Editorial**

José I. Gómez-Centurión

**Director Ejecutivo**

José M.º Díaz

**Redactor Jefe**

Juan José Martínez

**Diseño gráfico**

Fernando Chaumel

**Colaboradores**

Eduardo Ruiz

Javier Borceló

David Sopena

Robert Chatwin

Francisco Portalo

José A. Esteban

Cristino Gómez

Francisco Martín

Jesús Alonso

Pedra S. Pérez

Amalia Gómez

Alberto Suñer

**Secretaría Redacción**

Carmen Santamaría

**Fotografía**

Carlos Candel

Chema Socristán

**Portada**

J. Igual

**Ilustradores**

J. Igual, J. Pons, F. L. Frontán,

J. Septien, Pejo, J. J. Mara

**Edita**

HOBBY PRESS, S.A.

**Presidente**

María Andrión

**Consejero Delegado**

José I. Gómez-Centurión

**Jefe de Producción**

Carlos Peropadre

**Marketing**

Marta García

**Jefe de Publicidad**

Concha Gutiérrez

**Publicidad Barcelona**

José Galón Cartés

Tel: (93) 303 10 22/313 71 62

**Secretaría de Dirección**

Marisa Cogorro

**Suscripciones**

M.º Rosa González

M.º del Mar Colzoda

**Redacción, Administración y Publicidad**

Ctra. de Irún km 12,400

(Fuencarral) 28049 Madrid

Teléfonos: Suscrip.: 734 65 00

Redacción: 734 70 12

**Dto. Circulación**

Paulino Blanco

**Distribución**

Coedis, S. A. Valencia, 245

Barcelono

**Imprime**

ROTEDEC, S. A. Ctra. de Irún.

Km. 12,450 (MADRID)

**Fotocomposición**

Novocomp, S.A.

Nicolás Marales, 38-40

**Fotomecánica**

GROF

Ezequiel Solana, 16

**Depósito Legal:**

M-28468-1985

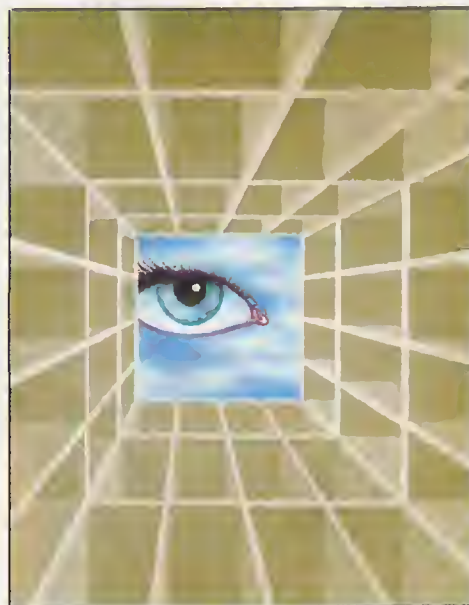
Derechos exclusivos  
de la revista**COMPUTING with  
the AMSTRAD**Representante para Argentina, Chile,  
Uruguay y Paraguay, Cia.  
Americana de Ediciones, S.R.L. Sud  
America 1.532. Tel.: 21 24 64. 1209  
BUENOS AIRES (Argentina).M. H. AMSTRAD no se hace  
necesariamente solidaria de los  
opiniones vertidas por sus  
colaboradores en los artículos  
firmados. Reservados todos los  
derechos.

Año 11 • Número 53 • 2 al 9 de Septiembre  
160 ptas. (incluido I.V.A.)  
Canarias, 155 ptas. + 10 ptas. sobretasa aérea  
Ceuta y Melilla, 155 ptas.

**Primeros pasos**

6

En el largo camino que uno recorre cada a cada con su ordenador, hay un momento que es conveniente, si no imprescindible, conocer un poco a fondo la trastienda de la máquina, de qué está hecha y cómo y por qué funciona. Este es el temo que ocupó esta semana a Primeros Pasos.

**Análisis**

10

El diseño de lenguajes son palabras mayores, y gran parte de las mejores programadores del mundo están ocupados en este menester. Sin embargo, la lógica es relativamente simple, y Análisis as da las rutinas de base para diseñar cualquier lenguaje interpretado.

**Código máquina**

14

El Amstrad destaca, entre otras muchas cosas, por sus magníficas cualidades de sanido. Para demostrar que éstas sirven para algo más que argumento publicitaria, vamos a aprender cómo exprimir las en lenguaje máquina.

**Mr. Joystick**

18

Hay juegos que hacen época. Movie es uno de ellos, una magnífica muestra de lo que puede ser la mezcla de un juego de aventuras con uno tipo Arcade, en sus justas proporciones y sacando partido de toda la que un Amstrad, en cuanto a gráficas, puede hacer.

**Serie Oro**

28

Rabin Hood no será nada comparada contigo, después de un poco de práctica con nuestro programa de Tiro con arco.

**Análogo**

32

Laga es recursivo, todo el mundo la sabe. Pero, cómo rastrear y depurar un procedimiento recursivo? He ahí la pregunta. En Análogo, la respuesta.



# micropc

P.º de la Castellana, 179-1.º  
28046 MADRID  
Teléf.: 442 54 33/44



## MERCENARIO

Cada pantalla es un nuevo desafío, una misión suicida, una dura batalla. Te encontrarás subiéndote frente a compañías de asalto, aviones, morteros, bombas de mano, etc. Hace falta mucho sangre fría y mucha agilidad mental para salir con vida.  
P.V.P. CASSETTE 2.000 Ptas.+I.V.A.  
DISCO 2.900 Ptas.+I.V.A.



## OBSIDIAN

La estación espacial Obsidian, construida en el interior de un asteroide se precipita hacia un agujero negro. Tú eres la única persona a bordo con el suficiente valor y experiencia para reactivar los sistemas de energía, arrancar los motores y salvar a tu tripulación.  
P.V.P. CASSETTE 1.900 Ptas.+I.V.A.



## PUZZLES

Por fin unos puzzles donde no se pierden piezas. Un reto a tu capacidad y a tu imaginación espacial. Un auténtico rompecabezas.  
P.V.P. DISCO 2.900 Ptas.+I.V.A.

## MICROSOFT

### Spitfire 40



## SPITFIRE 40

Trasladate a los años 40, Segunda Guerra Mundial, ponte a los mandos de un Spitfire. Siente la sensación de volar y combatir en aquellos aparatos donde la pericia del piloto era el 90% del éxito.  
P.V.P. CASSETTE 2.200 Ptas.+I.V.A.  
DISCO 2.900 Ptas.+I.V.A.

## JUMP JET



## JUMP JET

Estás en la cubierta de un portaaviones, debes despegar verticalmente en tu Harrier y, de repente, te encontrarás sólo en el aire, sin ver otra cosa que agua hasta que aparezcan los aviones enemigos dispuestos a destruirte.  
P.V.P. CASSETTE 2.200 Ptas.+I.V.A.  
DISCO 2.900 Ptas.+I.V.A.



## VIAJE FIN DE CURSO

Tomy ha acabado el curso y se quiere ir de vacaciones, pero antes debe pasar por todas las aulas y laboratorios para recoger sus calificaciones. El colegio es muy grande y, si no le ayudas, Tomy se quedará sin viaje fin de curso.  
P.V.P. DISCO 2.900 Ptas.+I.V.A.



## OLIMPIA PCW SHOW: PRIMER CONTACTO

Una vez más se ha celebrado en Londres, del 3 al 7 de septiembre, el Personal Computer World Show, PCW Show para los amigos, donde se reúnen, de una sola vez, las novedades más importantes del año en el terreno de los pequeños ordenadores.

El PCW Show es uno de los acontecimientos informáticos más importantes del mundo. **AMSTRAD SEMANAL** ha estado allí, en persona, para traerles a ustedes una semblanza en directo de las importantísimas novedades que se han mostrado en Inglaterra. En nuestra prisa por presentar a nuestros lectores información de máxima actualidad, vamos a concentrarnos esta semana en contarles un breve resumen del PCW Show; la próxima semana podrán leer un artículo mucho más extenso y detallado, comentando las incidencias de la Feria una por una.

El recinto ferial se hallaba dividido en dos partes: una dedicada a negocios, y, por tanto, a ordenadores que poseen esta filosofía, y la otra a máquinas que pueden incorporar ambos enfoques: el profesional y el lúdico, dando preeminencia a éste (con una notable excepción, que comentaremos luego).

En esta segunda parte es donde estaba el «meollo» de la cuestión. La asistencia, de público y firmas, era multitudinaria. Allí se encontraban las casas de software más importantes del mundo, y muchas marcas de ordenadores muy conocidas por nuestros lectores.

Pero vamos, sin más preámbulos, a las auténticas «bombas» de la Feria.

En primer lugar, el que va a ser sin duda el rey de los ordenadores de juegos de este año: el nuevo **Spectrum 2**. Esta excelente máquina, partiendo de la idea de Sinclair, ha sido refundida por **Amstrad**, y se nota su sello. Con respecto al **Spectrum Plus**, ha sido notablemente mejorado. Posee 128 kbytes de RAM, un diseño exterior nuevo, un magnífico teclado profesional y considerables mejoras en el campo de software y las

comunicaciones con el exterior, además de la novedad más importante y una gran idea: **lleva cassette incorporado**. Por supuesto, la compatibilidad con todos los programas anteriores del Spectrum está asegurada, y en el stand se podía ver corriendo en el **Spectrum 2** algunos de los juegos más populares del mercado. Todo esto, junto con su bajo precio, le convierte en una seria opción para todo aquel que necesite un ordenador para divertirse, sin desechar la posibilidad de alguna que otra aplicación de gestión: no hay que olvidar que estamos hablando de una máquina con 128 K y un Z80, potente y rápido.

La segunda bomba es, naturalmente, un **Amstrad**. Sugar ya nos tiene acostumbrados a las novedades, pero lo que ha hecho esta vez justifica plenamente el slogan publicitario de la compañía: increíble.

Les estamos hablando de un ordenador **Compatible con el IBM PC**, que posee las siguientes características grosso modo: 512 K de RAM, sistema operativo MS-DOS 3.2, procesador 8086 a 8 Mhz (dos veces más rápido que el del IBM), teclado profesional, unidad central con al menos un disco incorporado de 5¼ pulgadas, monitor con una resolución de 600 x 400 puntos, tarjeta de gráficos y color (opcional) y, atención: **ratón y sistema operativo GEM incluido**. Ya hemos hablado antes del GEM, y, aunque lo haremos más extensamente la próxima semana, baste decir que es un sistema basado en iconos y ventanas, que convierte al ordenador en algo muy fácil de usar, ya que no hay que recordar en absoluto complejas series de órdenes tipo CP/M. Pero lo que es verdaderamente increíble es el precio: 399 libras, es decir, a cambio del sábado 6 de septiembre, **96.957 pesetas**. Para que se hagan una idea, un sistema de prestaciones comparable proveniente de IBM saldría por aproximadamente **500.000**, sí, medio millón de pesetas.

La alternativa de **Amstrad** se da en distintas configuraciones, y el precio de la más cara, con monitor a color y disco duro de 20 Megas, cuesta 995 libras en Inglaterra (**241.785 pesetas**, también al cambio del 6 de septiembre). Por ahora, desconocemos el precio en España de la nueva máquina, que, como nuestros lectores ya saben, se presentará oficialmente en España el día 15 de septiembre, en el **Sonimag**.

Alguien dirá, con toda razón, que

# Primera PLANA

qué pinta esta máquina en la zona de «jueguecitos». Bien, nada; pero Sugar no es tonto, y sabía por anticipado que en ese lugar la afluencia de público sería mucho mayor, que también se pasarían por allí «los de negocios», aunque sólo fuera por curiosidad, y, por último, nada más entrar al Olympia, uno se daba de bruces con el stand de **Amstrad**, irremediablemente. De aquí la excepción, que no está muy claro si confirma o no la regla, pero tanto da.

## CUATRO NUEVOS PROGRAMAS DE MASTERSOFT

**MASTER-RADAR:** Es un juego basado en un Brazo Robot y en los distintos montajes que se pueden construir con él. Se trata de la pantalla de radar para seguimiento de aviones y helicópteros. Con ella debemos localizar los aviones que atacan nuestra base y destruirlos antes de que nos disparen a nosotros. 6128; 2.900, 464; 1.900.

**MASTERBINGO:** Edita cartones, extracciones de bolas manual o automático, listado de premios y comprobación. Precios: 6128; 2.900. 464; 1.900.

**MASTER-RULETA:** Es tan real que usted se encuentra envuelto en el casino de Montecarlo. Precios: 6128; 2.900. 465; 1.900.

**MASTERFACT:** Programa de mecanización del proceso de facturación y presupuestación y control de almacén. Con él puede controlar 200 clientes, 1.000 artículos y 1.500 apuntes de albarán y confeccionar, facturas y presupuestos por impresora. Precios: 8512; 19.900. 8256; 19.900.

Para más información diríjase a: **MASTERSOFT**  
Centro Comercial Sto. Domingo.  
Ctra. Burgos km. 28.  
28120 MADRID. Tel (91) 622 12 89.

# CONOZCA SU AMSTRAD A FONDO

*Todos los microprocesadores existentes en el mercado funcionan gracias a la unión de dos conceptos bastante diferenciados entre sí. Por un lado tienen una parte «eléctrica», o electrónica, que es la que se encarga de ejecutar el trabajo, y por otro necesitan de «algo» que indique a la anterior cómo, cuándo y qué es lo que tiene que hacer: es la componente «lógica».*



Pues bien, en esta ocasión vamos a desviarnos un poco de nuestra línea habitual, sin olvidarla por supuesto, y nos adentraremos en nuestro **Amstrad** para conocer un poquito sobre su parte física acompañándola de alguna que otra cosa más.

¿Qué le parece si comenzamos intentando explicar para qué sirve un ordenador?

De momento vamos a decir que nuestro **Amstrad** es una máquina que nos permitirá **tratar** correctamente, al menos eso esperamos, cualquier tipo de información. Y debido a que con un ordenador seremos capaces de manejar, en el buen sentido de la palabra, la información es por lo que a la **ciencia**, o como lo quieran nombrar, que recoge todo lo relacionado con estos temas se la llama **Informática**.

## Tratamiento de la información

Ahora bien, ¿a qué nos referimos cuando decimos **tratar la información**? Supongamos que vamos comi-

nando tranquilamente por nuestra ciudad y llegamos a un cruce de calles que está regulado por un semáforo. Lo primero que hacemos es mirar de qué color está y nos lo encontramos, por ejemplo, en verde.

En este instante estamos recibiendo **información**, ¿o no? Mientras estamos caminando llega a nosotros una serie de datos que percibimos por los ojos: el color del semáforo en este caso.

Y, ¿qué hacemos con ellos? Pues muy sencillo, los almacenamos en nuestra **memoria** para después analizarlos, compararlos, usarlos o cualquier otro cosa que se nos ocurra.

Sabemos, o por lo menos debíamos hacerlo, que cuando estemos ante un semáforo verde podemos cruzar la calle sin ningún problema. Este es un dato que ya teníamos almacenado antes de producirse la situación actual.

Si comparamos la información recibida **semáforo en verde**, con la que ya conocíamos del código de circulación, podemos deducir fácilmente que nuestro camino está libre, podemos seguir con nuestro paseo y el cerebro envía a los músculos las órdenes necesarias para que así sea: es el resultado obtenido.

Pues bien, de esto es de lo que hablamos al lanzar, así de sopetón, las palabras **tratamiento de la información**. Imaginamos que después de este ejemplito tendrá bastante más





claro de lo que se trata y podrá asociar fácilmente este nuevo concepto al campo informático.

## Input y Output

Nuestro ordenador realizará una lectura de datos a través de cualquiera de los órganos que tienen dispuestos para ella: teclado, disco, cinta, etc. Ahora bien, le están llegando una gran cantidad de ellos y de muy distinto tipo: no puede procesarlos todos. Así que sólo tomará los que le interesen y serán éstos los que trate. Imagine un juego de arcade en el que se utilizan las teclas de los cursores para el movimiento. Podremos pulsar cualquier otra tecla, pero es evidente que el ordenador la ignorará.

A continuación guardará el dato que le hemos dado en la memoria, bien en una variable, o en cualquier posición directamente. Es lo que está haciendo cuando ejecuta una instrucción del estilo de:

```
INPUT "EDAD", edad
```

El valor que nosotros le estamos dando desde el teclado lo introduce o almacena en la variable **edad** y después ¡ya verá lo que hace con él!

El siguiente paso sería comparar esta información con la que ya existiese dentro del ordenador y realizar la operación de acuerdo con los resultados obtenidos con este análisis. Sería como encontrarnos con una línea de programación que dijera algo como:

```
IF edad < 16 THEN  
PRINT "ERES MUY JOVEN"
```

Este es el resultado obtenido que nos aparecerá en la pantalla.

¿Cómo hace todo esto nuestro **Amstrad**? Hay una cosa que no debemos olvidar: un ordenador es, ante todo, una máquina o dispositivo electrónico y, por tanto, funciona a base de señales eléctricas pero un tanto especiales.

Tienen la particularidad que sólo pueden tomar dos valores predeterminados **cero** y **uno**. Por su similitud con el sistema de numeración binario —o de base dos— que sólo tiene dos dígitos diferentes, *el cero y el uno* es por lo que se les llama señales **binarias** o **digitales**. Más adelante veremos con más detalle las características de este nuevo tipo de numeración.

# Primeros PASOS

Pero tenemos una pregunta en el aire a la que hay que contestar, así que, ¡al grano! Nuestro ordenador dispone de unos componentes que se van a encargar de realizar cada una de las operaciones que forman parte de un **proceso** de información de datos. La figura 1 nos muestra un diagrama de bloques en el que están reflejadas cada una de las partes vitales del **Amstrad**.

Para introducirnos en la memoria del ordenador empleamos las llamados **dispositivos de entrada**, cuya función es traducir lo que le queremos decir a señales eléctricas que el **Amstrad** sea capaz de entender. Los más importantes y conocidos son el teclado y los **joysticks** tan necesarios, como todos sabemos, para poder indicar al micro la dirección de desplazamiento de nuestra nave hiperespacial o el momento justo para poder eliminar a cualquier alienígena que se interponga en nuestro camino.

Además de éstos, también necesitaremos otro tipo de unidades que se encarguen justamente de lo contrario. Del ordenador salen unos resultados que en un principio las humanas no somos capaces de entender. Por eso será necesario transformarlas en otras que ya sean legibles bien por nosotros o por algún elemento que se haya empleado para almacenar una serie de datos que luego utilizaremos en cualquier otro momento.

Seguro que ya habrá adivinado que estamos hablando de los llamados dispositivos de **salida**, entre los que se encuentran la pantalla, que nos visualiza los resultados obtenidos, la impresora, que puede proporcionarnos los reproducidos sobre un papel, y el altavoz de sonido que hace que el **Amstrad** pueda **hablar** con nosotros o emitir cualquier efecto especial que ponga la guinda a nuestro juego favorito.

Y hay otros que cumplen con las dos funciones: traducen la información en ambos sentidos. Son los **periféricos de entrada/salida**. Nuestro ordenador tiene un montón de posibilidades en este campo, ya que dis-



pone de unidades de disco o de cassette incorporadas al mismo equipo, dependiendo de los modelos y, además, está dotado de terminales que nos permitirán unirlo perfectamente a una segunda unidad de disco, o un cassette accesorio y a muchas variantes más. Es una joya.

Bueno, ya conocemos la manera de enviar datos o sacar resultados de nuestro **Amstrad**. Pero no se reduce a esto todo el proceso que se realiza en su interior, ¿verdad?

### La trastienda del Amstrad

Habíamos quedado que la segunda fase dentro de un **proceso** de tratamiento de información, consistía en almacenar los datos recogidos por las unidades de entrada; ya *podemos hablar en estos términos*. Para ello deberá poseer un elemento que sea capaz de hacerlo mientras se esté ejecutando un **programa** que los utilice.

Y ya que hemos añadido a nuestro vocabulario un nuevo término, el de programa, nos surge la idea que éste tendrá que estar ya dentro del ordenador para que pueda analizar los nuevos datos que hemos añadido, tratarlos convenientemente y devolvernos los resultados de este proceso.

De todo esto se encarga la **memoria**. Como vemos su función es múltiple ya que puede **almacenar** la información que le proporcionan los **periféricos** de entrada y además a su vez suministra los resultados obtenidos a la pantalla, impresora, etc., para que nos los visualicen o los guarden para otra posterior utilización.

Pero eso no es todo. También están contenidas en ella datos del sistema, que son los que van a conseguir que nuestro **Amstrad** funcione correctamente, las instrucciones que den forma a todos nuestros problemas, sus resultados intermedios y finales. Como ven, su trabajo es variado y fundamental por lo que será una de las partes del ordenador cuya investigación y análisis intentaremos ver con más detenimiento.

Otro de los componentes fundamentales dentro de un micro es el que hace todos los cálculos tanto aritméticos como lógicos. ¿Vemos su funcionamiento?

En la memoria tenemos bien guardados una serie de datos, ¿no es así?

Este elemento, al que llamaremos de ahora en adelante Unidad Aritmética Lógica **UAL** o Unidad de Cálculo, cogerá los que necesite, operará con ellos realizando cualquiera de las operaciones básicas aritméticas o lógicas y obtendrá un resultado. Pero no se queda con él, sino que lo envía de nuevo a la memoria y no a un lugar cualquiera de la misma, sino al que cada dato o variable tiene asociado.

De esto se desprende que un dato tiene que estar perfectamente localizable, o sea, que el micro conozca en todo momento el sitio exacto de la memoria donde está colocado. Añadamos entonces un nuevo concepto a todos los que ya poseemos: todos los datos, variables, etc., tienen asociada una **DIRECCION** mediante la que siempre los tendremos localizados y que será precisamente la posición que ocupan dentro de la memoria.

### Su majestad la CPU

Bueno, ya conocemos quién se encarga de los cálculos. Es hora de presentarle el elemento que controla o dirige todas las operaciones anteriores. Aquí la Unidad de Control, aquí un amigo. Es la parte vital del ordenador.

¿Cómo funciona? Pues simplemente indicando a cada uno de los demás componentes qué es lo que debe hacer y en qué momento debe actuar. ¿Qué le parece? Veámoslo un poco más despacio.

La Unidad de Control irá hasta la zona de la memoria donde se encuentre almacenado un programa. Allí tomará una por una las instrucciones que lo forman, las analizará

e irá asignando trabajo al resto de los elementos dependiendo del tipo de instrucción que vaya a ejecutar.

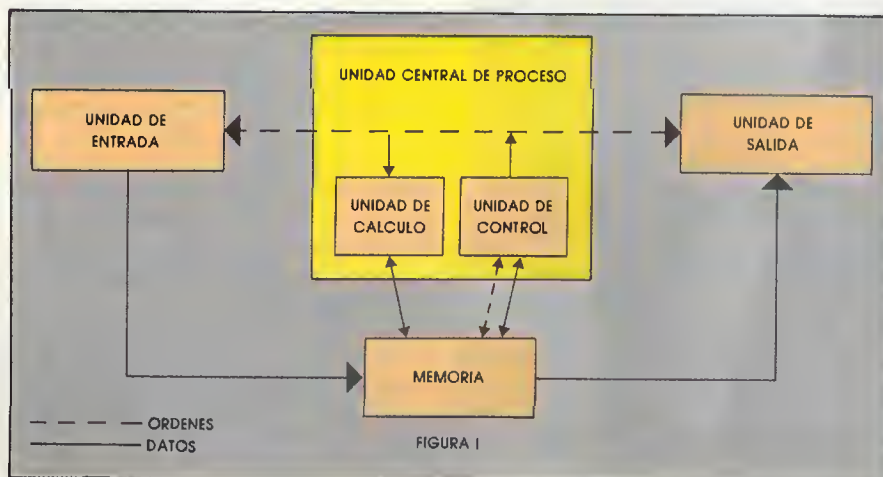
Pero éstas no son las grandes instrucciones en lenguaje Basic, o cualquier otro, a las que estamos ya acostumbrados sino que se trata de pequeñas órdenes elementales que realizan acciones muy concretas: sumar dos valores, por ejemplo.

Aquí está el primero de los problemas que presenta el ordenador. No es capaz de entender el lenguaje humano y sólo podemos comunicarnos con él utilizando el suyo propio. Pero recuerde que al fin y al cabo es una máquina eléctrica que solamente entiende un lenguaje compuesto exclusivamente por ceros y unos, **sistema binario**, agrupados convenientemente y que cada uno de ellos representará un determinado dato o instrucción.

Estaríamos ante un lenguaje de programación formado por números que el ordenador reconoce como los **códigos** de cada operación. Así que de momento tendremos que traducir toda la información que vaya destinada a ser tratada por el ordenador a este tipo de códigos, para que pueda captarla al detectar la presencia (1) o la ausencia (0) de una corriente eléctrica que pasa por un determinado circuito electrónico de los que componen el **Amstrad**. Es lo que llamamos **lenguaje máquina**.

### Comunicación hombre-máquina

Pero este tipo de lenguaje no nos aporta ningún género de información ya que generalmente no vamos a conocer la correspondencia existente entre estos códigos y sus res-





# Primeros PASOS



pectivas operaciones. Debido a esto, surgió entre los programadores la idea de emplear otros códigos que no fueran numéricos, sino que se parecieran más a la operación que queremos realizar.

Consistiría en dar a cada una de las instrucciones elementales un **nombre simbólico** que reflejara más claramente lo que con ella va a ocurrir, al menos para nosotros. Pero no olvidemos una cosa fundamental: el ordenador sólo conoce y entiende los programas codificados en lenguaje máquina.

Así que hemos añadido un trabajo extra al proceso ya que todo lo que codifiquemos en un lenguaje diferente tendremos que traducirla al código máquina equivalente. Esta labor podremos hacerla nosotros normalmente, pero es bastante obvio que nos resultará mucho más fácil si disponemos de un programa **traductor** que se encargue de ello. Y además mucho más fiable, ya que le podemos asegurar que nuestra memoria tiene algún que otro fallo.

Este tipo de lenguaje es el llamado **ensamblador** o de baja nivel y

resulta una interesante aproximación y entendimiento entre la forma de trabajo y comunicación del ordenador y los seres humanos.

Pero los programadores suelen ser unos personajes inquietos y no pueden estar parados. Empezaron a pensar en la forma de sustituir un grupo de estas instrucciones elementales por otro de más alto nivel que las abarcara. Y así nacieron los lenguajes de **alto nivel**.

Basic, Pascal, Fortran, Cobol, etc., son los nombres de alguno de ellos. Emplearlos supone innumerables ventajas ya que sus **palabras clave**, a nombres de instrucciones son bastante parecidas a las que nosotros usamos habitualmente, o al menos, a las que se utilizan en los países de habla inglesa. Por tanto, los programas codificados en alguno de estos lenguajes son bastante más fáciles de comprender y seguir que una realizada en **ensamblador**.

Además de agrupar cada una de las instrucciones de alto nivel, varias de bajo nivel, el número de las mismas es sensiblemente inferior por lo que el riesgo de error es ostensible-

mente menor. Imaginamos que entenderá el porqué, ¿no?

El **Amstrad** tiene incorporada en su sistema operativo un programa que se encargará de traducir una de estos lenguajes de alto nivel —el Basic— a código máquina que, no lo olvide, es el único que reconoce el micro.

Pero no lo hace con el programa completa, sino que va **interpretándolo** poco a poco, conforme se vaya ejecutando.

La forma de conseguirlo es ir consiguiendo instrucción por instrucción Basic, pasarla a código máquina y ejecutarla individualmente, pasando después a la que siga dentro del camino lógico que tenga que recorrer el programa. Y así tantas veces como sea necesaria.

¿Imagina lo que ocurriría al ejecutar un bucle FOR...NEXT que tenga que repetirse muchas veces?

El ordenador tendrá que **interpretar** y ejecutar el cuerpo del mismo tantas veces como lo indique la variable de control, gastando bastante tiempo en hacerla. Así pues, un programa interpretado puede resultar desesperantemente lento en su ejecución. Menos mal que el **Amstrad** pasee un Basic que es bastante rápida y por tanto podremos emplear este lenguaje para cualquier programa en el que se necesite una velocidad de ejecución relativamente alta.

Bueno. Creemos que habrán quedado ya suficientemente claras las dos componentes de las que consta un ordenador. Los circuitos electrónicos, elementos físicos, transistores, circuitos integrados, etc. que se encargan de realizar todas las operaciones que hemos descrito anteriormente es lo que llamamos **HARDWARE**.

A los programas, inmateriales y lógicos, que indican a los componentes eléctricos cómo y cuándo deben funcionar, los han bautizado con el nombre de **SOFTWARE**.

Unidas íntimamente, tal como lo están en nuestro **Amstrad**, nos permitirán realizar casi todo lo que se nos esté pasando por la imaginación. Súbase en el tren del **ingenio** con tan buena compañía y, ¡adelante!



*Suponemos que nuestros lectores se habrán preguntado en más de una ocasión cómo está hecho un lenguaje como el Basic, cómo funciona. Vamos a tratar de responder a esta pregunta a un nivel muy elemental, y lo que sigue se puede aplicar a cualquier lenguaje interpretado, como Logo, o el que se nos ocurra inventar.*



En lenguaje interpretado es aquel cuyos comandos u órdenes se van ejecutando uno a uno, a medida que el ordenador los lee de la entrada de datos, del **input**.

La tarea, muy resumida, podría dividirse en tres pasos:

- a) obtención de la palabra clave o token
- b) reconocimiento de que lo leído es efectivamente un token
- c) ejecución de la rutina correspondiente a dicho token

Visto esto, se desprende que la rutina clave del programa es la que abarca las líneas 190-280, la cual, a su vez, llama a otras dos, las de las líneas 140-170 y 80-120.

El método escogido es completamente general. La rutina **ObtenCaracter** toma una línea de **input** (por eso y otra razón que se verá luego se ha usado la orden **LINE INPUT** EN LA línea 20) y asigna a la variable **char\$** el carácter que marque el valor de la variable **col**. Como cada vez que se llama a esta rutina el valor de **col** aumenta en una unidad, nos **recorremos** toda la variable carácter a carácter, de uno en uno.

El carácter así obtenido se guarda en la variable **char\$**, sea alfabético, numérico o un signo menos, para permitir en nuestro lenguaje números positivos y negativos. Mientras estas condiciones se cumplan, los caracteres se acumulan en la variable **token\$** línea 240, que al final tendrá la palabra clave, o token, de nuestro lenguaje.

En la mayoría de los interpretados,

los token se separan unos de otros mediante espacios en blanco, como en el caso del Basic AMSTRAD, por lo que es necesario ejecutar la rutina de las líneas 130-170, **SaltaEspaciosBlancos**, ANTES del resto de **ObtenToken**. También los tabuladores se ignoran.

La rutina de las líneas 300-330, aunque el programa principal no la llama, está ahí para mostrar cómo se obtendrían los números cuando fuera necesario.

La ventaja de este método estriba, como dije antes, en su generalidad, porque también funcionará con ficheros de texto de disco. Para ello, bastaría transformar las primeras líneas del programa de la manera siguiente:

```
10 CLS: char$=""
11 OPENIN "NOMBRE"
12 WHILE NOT EOF
20 LINE INPUT #9, frases$
30 WHILE char$ < > ""
40 GOSUB 190: PRINT "Ejecutando"; token$
50 WEND
51 CLOSEIN
60 END
```

Resto del programa.

Es decir, tenemos la base no sólo de un intérprete, de cualquier intérprete, sino también de cualquier compilador. Para hacer con este programa algo espectacular, invéntense unos cuantos comandos, y pongan una línea del tipo:

ON lo—que—sea GOSUB...

siempre que **lo—que—sea** esté relacionado de alguna forma con el valor de **token\$**, y según ese valor, se ejecute una rutina u otra, adecuada a cada comando de los que se acaba de inventar.





# TU PROGRAMA DE RADIO

claro!



Audio 2

- Entrevistas a fondo
- Exitos en Soft
- Noticias en Hard
- Concursos

Programátelo: Sábados tarde de 5 a 7 horas.  
En directo y con tu participación.

**LA COPE A TOPE.**

— RADIO POPULAR 54 EMISORAS O.M. —

En Barcelona Radio Miramar





# FUNCION... ANDO CON NUMEROS



as facilidades que ofrece el **Amstrad** para operar aritméticamente son enormes. Hoy vamos a comenzar viendo la división entera. Si signo es: «/». Introduzca:

Cociente = 5/2  
PRINT cociente

Nuestra máquina nos da como respuesta ¡2!, ¿por qué? La división entera calcula el cociente entero e ignora el resto, por eso la respuesta no es 2.5.

Vamos con otra función: Módulo o MOD, se encarga de calcular el resto de la división entera de dos números o expresiones enteras. Puede comprobarlo con:

Resto = 22 MOD 7  
PRINT resto

después teclee **«Enter»** y la solución es el esperado 1. Experimente estas dos operaciones variando el dividendo y el divisor y analice los resultados.

Ahora, con nuestra recién conocida función MOD intentaremos actualizar la colegial prueba de la división, ¿recuerda?: cociente \* divisor + resto = dividendo.

## Programa uno

A partir de la línea 200 hacemos un tratamiento de un error con comandos que ya nos resultan familiares. Es muy importante realizar este tipo de comprobaciones para no llevarnos sorpresas desagradables. Estúdiese el programa, ¿qué dividendo y qué divisor originan cada tipo de error?

Y hablando de sorpresas, vamos a ver tres funciones que nos van a convertir números decimales en enteros. Teclee:

PRINT FIX (2.9857)  
PRINT INT (2.9857)

Ambas nos dan el mismo resultado 2, es decir, se toma el entero y se ignoran los decimales. Al menos aparentemente, parece que tenemos dos comandos que hacen lo mismo. Pero no es así, teclee ahora:

PRINT FIX (-2.9857)  
PRINT INT (-2.9857)

y nos encontraremos que mientras la primera devuelve -2, la segunda da como resultado -3. Esto nos aclara las diferencias. FIX devuelve la parte entera e ignora los decimales, mientras que INT redondea devolviéndonos el número entero inmediatamente inferior, en este caso -3. Podemos generalizar estos resultados y decir que cuando usamos números positivos los resultados de ambas funciones son los mismos, mientras

que en el caso de los decimales negativos la diferencia es de una unidad.

La tercera función que transforma números decimales en enteros es CINT. Su misión es redondear el decimal para conseguir el entero más próximo, tanto si es por exceso, como:

PRINT CINT (1.9872)

que devuelve 2, como por defecto:

PRINT CINT (1.2479)

nos saca un 1.

El programa 2 es un ejemplo, comparativo del trabajo de estas tres funciones.

## Programa dos

Por último nos encontramos con ROUND. Esta función nos permite redondear decimales y además determinar su precisión. Es decir, podemos convertir un número decimal en otro que tenga la cantidad de decimales que nosotros le indiquemos. Vamos a un ejemplo:

PRINT ROUND (3.1415965,4)

que nos dará nuestro conocido PI con cuatro decimales. Como puede ver esta función tiene dos parámetros, el primero es la expresión numérica a redondear y el segundo es el que señala la cantidad de decimales que queremos obtener. Como «deberes» vamos a dejarle con un par de ideas: ¿qué pasa cuando el segundo parámetro es un número negativo? **¿Y cuándo es 0?**

Siguiendo con los números vamos a cambiar el tema. El ordenador fun-



# Primeros repasos

```

10 REM PROGRAMA I
20 ON ERROR GOTO 200
30 CLS
40 INPUT "Dividendo ",dividendo
50 INPUT "Divisor ",divisor
60 cociente=dividendo\divisor
70 resto=dividendo MOD divisor
80 PRINT"Cociente="cociente,"Resto="
  resto
90 PRINT
100 PRINT"PRUEBA:"
110 PRINT"-----"
120 prueba=cociente*divisor+resto
130 PRINT"Cociente*Divisor+Resto=Di
  videndo"
140 PRINT
150 PRINT cociente*"divisor"+"rest
  o"="prueba;
160 IF prueba = dividendo THEN PRIN
  T"    CORRECTO"
170 IF prueba<>dividendo THEN PRINT
  "    ERROR"
180 PRINT
190 END
200 PRINT"  ERROR: ";
210 IF ERR=6 THEN PRINT"OVERFLOW."
220 IF ERR=11 THEN PRINT"DIVISION P
  OR CERO."
230 RESUME 40

```

```

10 REM PROGRAMA II
20 CLS
30 INPUT "ESCRIBA UN NUMERO DECIMAL
  ",numerodecimal
40 resultado1=FIX(numerodecimal)
50 resultado2=INT(numerodecimal)
60 resultado3=CINT(numerodecimal)
70 PRINT
80 PRINT" FIX"," INT"," CINT"
90 PRINT" ---"," ---"," ----"
100 PRINT resultado1,resultado2,res
  ultado3

```

```

10 REM PROGRAMA III
20 CLS
30 INPUT "ESCRIBA UN NUMERO DECIMAL
  ",numerodecimal
40 PRINT
50 INPUT "CON CUANTOS DIGITOS? ",di
  gitos
60 PRINT
70 PRINT"BINARIO: "BIN$(numerodecim
  al,digitos)
80 PRINT"HEXADECIMAL: "HEX$(numerod
  ecimal,digitos)

```

ciona con sistema binario, es decir, que sólo conoce dos estados: 0 (falso) y 1 (verdadero), pero para nosotros escribir cualquier cantidad a base de ceros y unos, además de ser muy pesado, puede hacernos cometer errores fácilmente. Por esto, para dar mayor claridad a nuestra conversación con el **Amstrad**, vamos a utilizar la base 16.

Nuestra intención no es dar una «lección magistral» de sistemas de numeración, sino presentar las funciones que nos van a servir para pasar números decimales a sistema binario o base dos: BIN\$ y a sistema hexadecimal a base dieciséis: HEX\$. Observe que ambas acaban con el símbolo «\$», por lo que podemos in-

tuir que van a devolvernos cadenas.

Su sistema de trabajo es similar. Las dos admiten un segundo parámetro con el que podemos indicar al **Amstrad** cuantas cifras queremos que tenga nuestro número. Véalo gráficamente:

PRINT BIN\$(10,8)

nos devuelve una cadena de cuatro caracteres que forman la expresión binaria del número 10.

PRINT HEX\$(10,4)

nos devuelve una cadena de cuatro caracteres con el equivalente hexadecimal de 10. En los dos casos la longitud del segundo parámetro oscila entre 0 y 16, pero no es neces-

rio conocerla de antemano, ya que si damos una medida menor que la necesaria, nuestro ordenador no truncará la cadena. Pruébela con:

PRINT BIN\$(255,3)

Recuerde: estas dos últimas funciones devuelven cadenas. No las utilice para operaciones matemáticas. Le dejamos por el momento con un programa que utiliza nuestros últimos hallazgos. Es muy sencilla y no le planteará la menor dificultad.



# MUSICA EN LENGUAJE MAQUINA

*Empezaremos hoy con el estudio de otro de los grandes capítulos de que se compone la programación, tanto en Basic como en código máquina. Se trata de la producción de música o sonidos a través del ordenador.*



Como todos sabemos, el **Amstrad** tiene un comando Basic que nos permite generar cualquier tipo de nota musical o ruido, dicho comando, es el siguiente:

SOUND C,P,D,V,EV,ET,R

Dando diferentes valores a cada uno de los parámetros, para relacionarlos más tarde con los parámetros que utiliza la rutina del firmware que se encarga de generar los sonidos.

Indicaremos a continuación cuál es el significado de cada uno de los parámetros, podremos obtener cualquier tipo de nota o ruido que deseemos en cada momento.

Indicaremos a continuación cuál es el significado de cada uno de los parámetros, para relacionarlos más tarde con los parámetros que utiliza la rutina del firmware que se encarga de generar los sonidos.

C...SITUACION DE CANALES  
P...PERIODO DE TONO  
D...DURACION  
V...VOLUMEN INICIAL  
EV...ENVOLVENTE DE VOLUMEN  
ET...ENVOLVENTE DE TONO  
R...RUIDO DE FONDO

Para producir sonidos desde código máquina, utilizaremos la rutina que hemos mencionado anteriormente, que es la que se indica a continuación:

COLOCA UN SONIDO  
EN LA COLA DE SONIDOS. #BCAA





Añade un sonido a la cola de sonidos del canal que se indique. Si las colas están llenas, entonces no se produce ningún efecto.

#### Condiciones de entrada

El registro doble HL debe contener la dirección de datos de sonido.

#### Condiciones de salida

Si los datos se han logrado colocar en la cola de sonidos:

- HL se pierde
- Carry verdadero

Si la cola estaba llena:

- HL se preserva
- Carry falso

En cualquier caso se corrompen los registros A, BC, IX y DE y todos los flags.

Hemos dicho que el registro doble HL, debe contener la dirección de los datos del sonido que se quiere producir. Veamos cuáles son dichos datos y en qué orden se deben colocar. Apuntaremos también con qué parámetros del comando SOUND se corresponde.

## Código MAQUINA

Para efectuar dicho programa en código máquina, lo primero que debemos hacer es resetear la cola de sonido, para asegurarnos de que no sonará alguna nota que haya sido colocada anteriormente a la ejecución del programa.

Esto se logra mediante la llamada a la rutina del firmware que produce este efecto, y que es la siguiente:

**RESETEA COLAS DE SONIDO. #BCA7**

Inicializa las colas de sonido.

#### Condiciones de entrada

No precisa.

#### Condiciones de salida

Se corrompen los registros AF, BC, DE y HL. Los demás registros son preservados.

PARAMETRO DEL COMANDO SOUND DE BASIC	LUGAR QUE OCUPA DENTRO DE LA DIRECCION DE DATOS	PARAMETRO
Situación de canales	byte 0	Canal que se desea utilizar
Envolverte de volumen	byte 1	Numero de la EV a utilizar
Envolverte de tono	byte 2	Número de la ET a utilizar
Periodo de tono	bytes 3-4	Nota que se desea obtener
Ruido	byte 5	Ruido de fondo
Volumen inicial	byte 6	Amplitud inicial
Duración	bytes 7-8	Duración de la nota

De esta forma cada uno de los bloques de datos que conformen una nota, deberán poseer una longitud de 9 bytes, que podrán ser colocados en cualquier posición de la memoria, indicando dicha dirección en el registro HL antes de hacer la llamada a la anterior rutina.

El programa que hemos preparado para este capítulo, presenta el comando SOUND con una serie de parámetros que se varían en cada uno de los bucles de programa.

Se produce en primer lugar una variación del volumen de la nota, a continuación se da diversos valores a la duración de la nota, y por último se incorpora ruido de fondo al sonido que se va a producir.

Se realizan las siguientes operaciones:

- Se limpian las colas de sonido.
- Separa cualquier sonido que se esté produciendo.
- El generador de sonido se silencia.

Una vez hecho esto, se entra en un bucle en el cual se varía el parámetro que indica el volumen de la nota, y se llama a la rutina encargada de generar el sonido.

Seguidamente nos encontramos con un bucle similar al anterior, que en este caso varía la duración de la nota que se va a emitir, llamando así mismo a la rutina generador de la música.



## PROGRAMA BASIC

```

10 REM ****DIFERENTES VOLUMENES****
20 REM
30 DURACION=1:RUIDO=0
40 FOR VOLUMEN=4 TO 7
50 GOSUB 220
60 NEXT
70 REM
80 REM ****DIFERENTES DURACIONES***
*
90 REM
100 RUIDO=0:VOLUMEN=7
110 FOR DURACION=1 TO 3
120 GOSUB 220
130 NEXT
140 REM
150 REM ****DIFERENTES RUIDOS****
160 REM
170 DURACION=1:VOLUMEN=7
180 FOR RUIDO=1 TO 3
190 GOSUB 220
200 NEXT
210 END
220 FOR NOTA=1 TO 550
230 SOUND 1,NOTA,DURACION,VOLUMEN,,
,RUIDO
240 NEXT
250 RETURN

```

## PROGRAMA CARGADOR

```

10 REM *PROGRAMA CARGADOR*
20 FOR N=&A000 TO &A08C
30 READ A:SUMA=SUMA+A
40 POKE N,A
50 NEXT
60 IF SUMA<>&3A10 THEN PRINT "ERROR
EN DATAS"
70 DATA 205,167,188,62,1,50,138
80 DATA 160,175,50,136,160,62,4
90 DATA 50,137,160,6,7,197,205
100 DATA 88,160,58,137,160,60,50
110 DATA 137,160,193,16,242,175,50
120 DATA 136,160,62,7,50,137,160
130 DATA 62,1,50,138,160,6,3
140 DATA 197,205,88,160,58,138,160
150 DATA 60,50,130,160,193,16,242
160 DATA 62,1,50,138,160,50,136
170 DATA 160,6,3,197,205,88,160
180 DATA 58,136,160,60,50,136,160
190 DATA 193,16,242,201,33,0,0
200 DATA 34,134,160,1,30,2,197
210 DATA 42,134,160,35,34,134,160
220 DATA 205,115,160,193,11,120,177
230 DATA 32,239,201,62,1,205,173
240 DATA 188,230,7,40,247,33,131
250 DATA 160,205,170,188,201,1,0
260 DATA 0,0,0,0,7,1,0

```

## LISTADO DE ENSAMBLADO

Hisoft GENA3.1 Assembler. Page 1.

Pass 1 errors: 00

```

A000          10          ORG #A000
                20 ;
                30 ;RESETEA LAS COLAS DE SONIDO
                31 ;
                40 ;
A000 CDA7BC     50          CALL #BCA7
                60 ;
                70 ;DIFERENTES VOLUMENES
                80 ;
A003 3E01       90          LD A,1
A005 328AA0    100         LD (DURAC),A
A008 AF        110        XOR A
A009 3288A0    120         LD (RUIDO),A
A00C 3E04      130         LD A,4
A00E 3289A0    140         LD (VOLUM),A
A011 0607      150         LD B,7
A013 C5        160 BUC1:  PUSH BC
A014 CD58A0    170         CALL SONAR
A017 3A89A0    180         LD A,(VOLUM)
A01A 3C        190         INC A
A01B 3289A0    200         LD (VOLUM),A
A01E C1        210         POP BC
A01F 10F2      220         DJNZ BUC1
                230 ;
                240 ;DIFERENTES DURACIONES
                250 ;
                260        XOR A
A022 3288A0    270         LD (RUIDO),A
A025 3E07      280         LD A,7
A027 3289A0    290         LD (VOLUM),A
A02A 3E01      300         LD A,1
A02C 328AA0    310         LD (DURAC),A
A02F 0603      320         LD B,3
A031 C5        330 BUC2:  PUSH BC
A032 CD58A0    340         CALL SONAR
A035 3A8AA0    350         LD A,(DURAC)
A038 3C        360         INC A
A039 328AA0    370         LD (DURAC),A
A03C C1        380         POP BC
A03D 10F2      390         DJNZ BUC2
                400 ;
                410 ;DIFERENTES RUIDOS
                420 ;
                430         LD A,1
A03F 3E01      440         LD (DURAC),A
A041 328AA0    450         LD (RUIDO),A
A044 3288A0    460         LD B,3
A047 0603      470 BUC3:  PUSH BC
A049 C5        480         CALL SONAR
A04A CD58A0    490         LD A,(RUIDO)
A04D 3A88A0    500         INC A
A050 3C        510         LD (RUIDO),A
A051 3288A0    520         POP BC
A054 C1        530         DJNZ BUC3
A055 10F2      540         RET
A057 C9        550 ;
                560 ;VARIACION DE NOTAS
                570 ;
A058 210000    580 SONAR: LD HL,0
A05B 2284A0    590         LD (NOTA),HL
A05E 012602    600         LD BC,550
A061 C5        610 BUC:  PUSH BC
A062 2A86A0    620         LD HL,(NOTA)
A065 23        630         INC HL
A066 2286A0    640         LD (NOTA),HL
A069 CD73A0    650         CALL MUS1C
A06C C1        660         POP BC
A06D 0B        670         DEC BC

```





A06E	78	680	LD	A,B
A06F	B1	690	OR	C
A070	20EF	700	JR	NZ,BUC
A072	C9	710	RET	
		720 ;		
		730 ;	RUTINA	DE MUSICA
		740 ;		
A073	3E01	750	MUSIC:	LD A,1
A075	CDADBC	760	CALL	#BCAD
A078	E607	770	AND	7
A07A	28F7	780	JR	Z,MUSIC
A07C	2183A0	790	LD	HL,DATOS
A07F	CDAABC	800	CALL	#BCAA
A082	C9	810	RET	
		820		
A083	010000	830	DATOS:	DEFB 1,0,0
A086	0000	840	NOTA:	DEFW 0
A088	00	850	RUIDO:	DEFB 0
A089	07	860	VOLUM:	DEFB 7
A08A	0100	870	DURAC:	DEFW 1



## Código MAQUINA

o no espacios libres dentro de la cola de sonido.

Por lo tanto, la rutina se escribirá de la forma siguiente:

```
MUSIC: CALL #BCAA
AND 7
JR Z, MUSIC
```

De esta forma mientras la cola esté llena, no se intentará colocar una nueva nota en ella, ya que si los tres primeros bits del acumulador están a cero, indico que la cola está llena en estos momentos.

Una vez que se haya producido un hueco en dicha cola de sonido, se efectuará la transmisión de los nuevos datos, que se produce de la forma siguiente:

```
LD HL, DATOS
CALL #BB5A
RET
```

Cargamos el registro doble HL con la dirección donde se encuentran los datos, y llamamos a la rutina encargada de colocar dichos datos en la cola de sonido, para que el procesador de sonido la ejecute.

De esta forma se irán produciendo las diferentes notas que genera el programa, sin que ninguna de ellas se pueda perder a causa de un sobrellenado de las colas.

### Condiciones de entrada

El acumulador debe contener el bit correspondiente al canal que se desea observar.

```
BIT 0, CANAL 1
BIT 1, CANAL 2
BIT 2, CANAL 3
```

### Condiciones de salida

El acumulador contiene el estado del conal, en los bits 0,1 y 2 se indica el número de espacios libres. Los registros BC, DE y HL así como todos los flags se corrompen. Los demás registros se preservan.

Así, pues, llamando a la anterior rutina, podemos observar si quedan

Por último, se entra en el último bucle encargado de variar el ruido de fondo que se va a producir.

Vamos a ver a continuación cómo funciona la rutina encargada de producir el sonido «MUSIC».

Cuando se entra en esta subrutina, lo primero que hacemos es observar el estado de la cola que vamos a utilizar, en este caso la correspondiente al canal 1. Para ello utilizaremos una rutina del firmware que nos informa del estado de éstos.

### CHEQUEA LAS COLAS DE SONIDO. #BCAD

Observa el estado de las colas de sonido, indicando el número de espacios libres dentro de dichas colas.

## Correo..., más rápido...



Con el fin de acelerar lo más posible el **correo**, y poder resolver o contestar a todas las dudas y sugerencias que llegan a nuestra redacción, a partir de esta semana os rogamos, en beneficio de todos, consignar en el sobre, en lugar bien visible, una de las denominaciones siguientes:

- **Suscripciones AMSTRAD.** Para todos aquellos casos relacionados con petición de cintas, números atrasados, formalización de suscripciones, devoluciones etc...
- **Mercado Común AMSTRAD.** Compras, ventas, intercambios, clubs...
- **Sin duda alguna AMSTRAD.** Para que nos enviéis todas vuestras dudas.
- **Serie Oro AMSTRAD.** Para los programas que nos enviéis para su publicación.
- **Sugerencias AMSTRAD.** Para vuestras críticas, sugerencias o cualquier opinión que queráis vertir sobre la revista.



# MOVIE

**Sin saber cómo, nos hemos metido en una película de gánsters de la cual no podemos salir; una vía de escape es la muerte, la otra, el éxito, ¿qué nos deparará nuestra aventura?**



Un oscuro despacho de la decimaviena planta, de una de los colosales rascacielos que se erigen sobre el suelo de Nueva York, se encuentra la oficina del más famoso detective de la historia.

Estamos hablando ni más ni menos que del intrépido Jack Marlow, el luchador infatigable capaz de resolver, con la única ayuda de su ingenio las cosas más complicadas.

La puerta del pequeño despacho se abre y una persona enfundada en una gabardina tipo Colombo, se planta delante de la mesa de Jack.

Su voz, casi de ultratumba y ronca por los efectos de una vida de acérrimo fumador y bebedor de whisky, nos revela las palabras que nos pondrán en la pista del último caso que va a abarcar el inefable Marlow en su vida.

«Me envía mister... y el asunto que te conlleva es de vida o muerte, debes recuperar, cueste lo que cueste, la cinta magnetofónica que Bugs Malloy guarda en su caja fuerte, el contenido de la cinta ya ha acabado con la vida de una docena de personas y la cuenta no está cerrada.»

Cuando Jack se dispone a interrogar al inesperado visitante, un ruido de cristal roto hace que el rastro de su interlocutor se tense en una acusada mueca de dolor, inmediatamente el cuerpo cae pesadamente sobre el respaldo del sillón.

Sin esperar el segundo disparo, Marlow se arroja al suelo en una reacción felina y empuña su pistola esperando lo peor. Otras im-

pactos de las balas en la mesa y en el resto del mobiliario indican que el peligro sigue latente.

En un arrojado de valor, y catapultado por la sangre que le hierve en las venas, Marlow se arrastra a gatas hacia la puerta y sale de su despacho, refugiándose en el pasillo donde se sabe a salvo.



Unos minutos de espera, y la calma se restablece en su despacho, las sirenas de la policía suenan en la calle y parece que el peligro ha desaparecido.

Al entrar en su oficina, el fiambre que ocupa el sillón de las visitas, permanece con la mirada fija en la ventana por la que entró el tiro.

El resto de los objetos de la habitación continúan en orden, sólo unas cuantas agujeros



en la mesa y los sillones denotan que allí ha habido un tiroteo.

Son cerca de las 2 cuando la policía abandona el despacho, habiendo cumplido con las diligencias reglamentarias en estos casos.

Tras cerca de dos horas de preguntas, fotos, firma de atestados y demás trámites, Marlow se encuentra agotado, con poca decisión se dirige hacia una de los archivadores, extrayendo del cajón superior una botella de whisky.

Botella en mano se sienta en su silla, y llena un vaso que bebe de un solo trago, un par de vasos más y la sangre vuelve a circular por sus venas.

En su mente están grabadas las palabras de su visitante, en un pequeño destello, la imagen de Bugs Malloy viene a su imaginación, sumiéndole en una terrible confusión.

«Entrar en el despacho de Malloy y conseguir la cinta.»

Nada más y nada menos, ni un ejército la conseguiría, ¿que se cree, mister... que soy un superhombre?

La imagen de Malloy sigue indeleble en su cerebro; «sólo le he visto una vez y estaba rodeado de hombres armados, fue en el entie-







rrero de Luchio Petruhi, hace casi dos años, desde entonces no nos hemos vuelto a cruzar».

Bugs Malloy tiene sus oficinas situadas en un edificio de su propiedad, en el que las últimas cinco plantas están dedicadas a la administración de los negocios sucios que dirige.

Ni que decir tiene que el acceso a la zona de seguridad del edificio está protegido por hombres armados y que en las inmediaciones, guardaespaldas y gente del hampa se encargan de los visitantes inoportunos.

Para colmo, el despacho de Malloy se encuentra en la parte más alta del inmueble, y llegar hasta él puede ser una verdadera locura.



Dándole vueltas al asunto, recuerda un lique de la calle 63, una rubia platina de esas que hacen perder el aliento, que mira por dónde, trabaja en ese edificio.

«Ahora que recuerdo, debía de trabajar en la zona de seguridad, porque siempre que hablábamos de su trabajo las respuestas eran vagas y a veces incoherentes.»

Inmediatamente Marlow se cuelga del teléfono; «Tanya, no sé si me recuerdas, soy Jack»... «¿qué Jack?»... «¡El guapo, por su-

puesto! ¿Por qué no quedamos para cenar?»

Una cena romántica y una arrebatadora noche de pasión hacen maravillas (nuestro héroe no sólo es bueno con la pistola) y Tanya se compromete a guiarnos en el interior del edificio.

La película comienza para ti, estás en el edificio de Malloy, además desarmado, porque en el control de entrada se registra a todo el mundo, hazte con una pistola, busca a Tanya, consigue la cinta y llévala a tu despacho, ponla en el magnetófono y sabrás lo que ha llevado a la muerte a tu último visitante y a los otros doce.

Ya metidos en acción, la pantalla nos va desvelando las interioridades de esta «película». Fuera de toda duda nos encontramos ante un juego en el que la animación, gráficos y demás elementos visuales del mismo, responden a un patrón con un rancio abolengo en los programas de ordenador.

Sin duda, nos referimos al padre de las modernas técnicas de puesta en escena de juegos, el Knight Lore, todo un clásico cuya sombra se cierne sobre todo el software de calidad.

Cualquier juego que quiera realizarse sobre un decorado de habitaciones, en las cuales la comunicación se realiza en distintas direcciones, y dentro de las que queremos ver los ob-



jetos que hay, debe realizarse con esta técnica.

La perspectiva de los juegos realizados de esta forma favorece la confección de mapas por parte de los usuarios, y la orientación dentro del elevado número de pantallas que contienen.

Imagine en esta ocasión han decidido escoger la técnica adecuada, que aparte del Knight Lore, ha empleado programas como el Sweva's, Rasputin, Alien 8 y otros ejemplos destacados de los programas de búsqueda.

Pero la representación de los espacios en pantalla sólo ocupa la parte visual del programa; el resto, que constituye el guión del programa y su desarrollo en el tiempo, es dominio de la imaginación de los creadores, y aquí sí tenemos que decir que Imagine, ha hecho honor a su nombre.

Por primera vez en el mundo del software nos encontramos ante una aventura interactiva, en la que los distintos personajes tienen personalidad propia y con los cuales podemos hablar.

## Mister JOYSTICK

Desde luego, hemos visto muchas cosas en los programas de juegos, pero el hecho de poder entrar en una habitación y hablar con los tíos y tías que allí se encuentran, nunca habíamos tenido el placer de contemplarlo.

Los expertos en aventuras de texto, en las que a los pies de una pantalla estática aparece un texto y nosotros tecleamos otro, nos dirán que ellos ya lo habían experimentado, pero a esto no nos referimos, nos encontramos en una nueva dimensión.



En nuestra «película» estamos en una aventura animada, con personajes que se mueven, que nos atacan; con puertas, con cientos de habitaciones, y además con personajes con los que hablamos y a los que podemos sobornar, golpear, asesinar y demás.

Tratándose de un producto inglés, los personajes, por supuesto, hablan en inglés; por lo tanto, nosotros tendremos que expresarnos en inglés, pero no hay que alarmarse por el hecho.



Cuando nos dirigimos a cualquier personaje disponemos de cinco frases mágicas, que nos sacarán de cualquier aprieto, siempre, claro está, que esto sea posible.

Help # nos proporcionará cualquier ayuda disponible.

What is the password # nos ayudará a conocer la contraseña para superar las puertas con custodia.

Which way do I go # para elegir el camino correcto.



Whot do I do next # nos informará sobre lo próximo a realizar.

How much money do you want # imprescindible cuando queremos sobornar a alguien.

Los que sepan inglés, encima podrán intentar todas las frases y preguntas que se les ocurran, con lo que pueden conseguir algún efecto interesante.

Cuando alguien se dirige a nosotros, aunque no sepamos ni godo de inglés, podremos

entenderle; las instrucciones del juego contienen la traducción de las 58 frases, que constituyen el repertorio de preguntas, respuestas e indicaciones que nos puede dar cualquiera de los personajes del juego.

Con el listado de frases, podemos enterarnos perfectamente de lo que pasa y superar los problemas idiomáticos sin necesidad de tener un diccionario al lado.

Algunas de las frases utilizadas nos dan idea de la coherencia y la perfecta trama psicológica con que se ha elaborado el juego;

Haven't we meet before? # ¿No nos hemos visto antes?

Are you drunk or what? # ¿Estás borracho o qué?

Smart guy, ah ! # ¡Vaya tío listo!

You are going a great job # Adelante, lo estás haciendo muy bien.

Don't be silly # No seas tonto.

I'll kill you! # Te mataré.

Who do you think you are? # ¿Quién narices te crees que eres?



What do you expect me to say? # ¿Qué quieres que te diga, tío?

Try de second door # Prueba por la segunda puerta.

Leave the bag here # Deja aquí la bolsa.

What do you mean? # ¿Qué quieres decir?

Así hasta 58 frases que nos revelarán los más íntimos secretos de las recónditas dependencias del área de seguridad de las oficinas de la Malloy's Company.



¿Cómo se realiza el diálogo?, sólo nos queda decir que «de película». Cuando queremos hablar con alguien, o un personaje nos dirige la palabra, empezamos a comprobar la astucia de los autores del juego.

El diálogo se realiza con la más auténtica técnica del comic, al hablar un personaje, sobre su cabeza aparece la típica burbuja que contiene los diálogos en los comics, dentro de la cual encontramos el texto.

Si somos nosotros los que hablamos, nuestra burbuja surge y dentro de ella aparece el texto que tecleamos en el ordenador.

La cosa no acaba ahí, nuestro hombre puede realizar una amplia gama de tareas que le permiten pasar por las distintas habitaciones y moverse aprovechando todo lo imaginable.

Podemos saber todo lo que llevamos encima, con el inventario, podemos coger objetos y soltarlos, disparar nuestra pistola, utilizar nuestro puño para noquear a los intrusos, podemos hablar y hasta podemos arrojar cuchillos y demás armas blancas, la opción de andar cierra la lista.

La selección de lo que queremos hacer se realiza directamente con el joystick, sin tener que aprendernos un montón de teclas, para esto se ha utilizado una técnica iconográfica que facilita mucho las cosas.





En la parte inferior de la pantalla se encuentran el gráfico de actividades, en el cual esquemáticamente se representa cada una de ellas, de forma que con un toque de disparo podemos mover el cursor de actividades en la dirección requerida; fijado éste sobre la deseada, otro toque de fuego hará que Marlow la ejecute.

Este método tiene un pequeño inconveniente, que es que como andar es una actividad

más, mientras disparamos, golpeamos o tiramos algo, nuestro personaje permanece inmóvil, constituyendo un blanco fácil para los pistoleros que se encuentran en las habitaciones, los cuales sí que se mueven.

En este juego, como en todos los de su estilo, hay que prestar especial atención a los objetos que encontramos en cada habitación; en cualquier sitio podemos encontrar cuchillos, pistolas, dinero, whisky y demás aditamentos que emplearemos en nuestro periplo por el edificio.

Hay que destacar el cuidado y el estudio, hasta el más mínimo detalle, que se ha hecho de objetos, personajes y situaciones.

Al entrar en una habitación, existen objetos que podemos coger, otros que podemos utilizar, sillas y muebles que se mueven, etc.

Al coger una pistola, disponemos de seis balas, las cuales se van gastando cada vez que disparamos, haciendo inservible el arma tras el sexto disparo; el juego está estudiado en sus más mínimos detalles.

Los gráficos y el movimiento siguen fielmente la estirpe de su casta, están realizados con un dibujo tipo comic, en el cual, para acentuar el carácter dramático de la aventura (nos encontramos en la época de la ley seca, rodeados de matones sin piedad, cuya vida no vale ni medio dólar), se ha elegido un tipo de personajes alargados y de movimientos cadenciosos que nos meten de lleno en «La Película».

Los gráficos realizados en el modo de mediana resolución en cuatro colores, son de una nitidez y precisión que no permiten objeción alguna.

Dada la complejidad de la distribución de objetos en las habitaciones, y conscientes de que conducir a nuestro personaje con un sistema de cuatro sentidos distintos de movimiento no es tarea fácil, se han previsto cuatro modalidades de movimiento.

Podemos elegir entre el control total del personaje, con el que indudablemente chocare-



## Mister JOYSTICK

mos con multitud de sillas, percheros y demás; o elegir la esquivia automática de objetos, con la cual Marlow pasará a su lado sin siquiera rozarlos.

También podemos usar el método rotación estática-dirección de movimiento o el de dirección de movimiento simple.

Con el primero giraremos sobre nosotros mismos hasta quedar en la dirección de movimiento deseada por la cual avanzaremos,

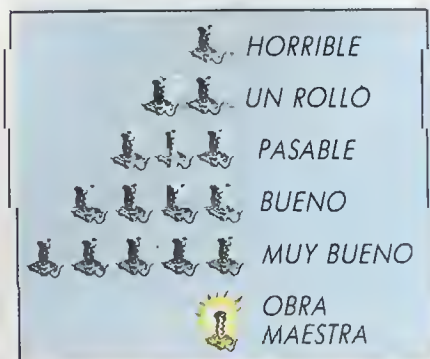


con el segundo cambiaremos directamente de dirección, sin giro estático.

Como se verá, en este programa hay de todo.

Movie es un programa de éstos que se ve uno muy de vez en cuando; estudiado hasta en sus más últimos detalles, nos ofrece simultáneamente acción, muerte, diálogo, emoción, incertidumbre y todo lo que rodea a una aventura en la que hemos entrado y no sabemos cómo vamos a salir. Tan sólo conocemos nuestra meta, apoderarnos de la cinta magneto-fónica, el resto debe averiguarlo un detective valiéndose de su valor e instinto.

Un programa de éstos que durante semanas y semanas mantiene viva nuestra atención, encontrando cada día que abordamos la aventura nuevas pistas y detalles que nos conducen hasta la meta final





# ORDEMANIA SOFT

DE VENTAS  
ESTADÍSTICA

**por fin..**

## GESPAC

Paquete integrado de gestión que le permite a Vd. de forma fácil y optimizando el tiempo, la gestión global de su empresa.

Este paquete incorpora los programas de CONTABILIDAD, FACTURACION, CONTROL DE STOCKS además de un programa de CONTROL DE PEDIDOS y todo de forma interactiva. Controle su empresa por sólo **29.900 pts.**

### Contabilidad

Contabilidad de fácil manejo y de gran potencia que permite trabajar con cuentas de hasta cuatro niveles, con capacidad según diskette de 500/1.000 cuentas y de 2.000/10.000 asientos.

Permite modificar o dar de baja apuntes ya integrados en el Mayor, programaciones de cierres, ficheros de Contabilidad y Cuenta de Explotación, ejecución de balances comparativos, reenlazándolos por meses, clave acceso restringido, etc...

**19.900 ptas.**

### Facturación

Programa de gran agilidad y rapidez que incorpora el Control de Clientes, con gran capacidad de datos, Artículos, Albaranes, Facturas y recibos.

Generación automática y manual de documentos, valoración, a voluntad de los albaranes, todo tipo de listado, incluido el del IVA de las facturas emitidas para la declaración de Hacienda, etc...

**15.500 ptas.**

### Control de stocks

Gran capacidad de datos, le permitirán a Vd. llevar con claridad y sencillez el control de su stock.

El programa le permite llevar un libro de entradas/salidas, reorganizarlo, hacer listados de stocks..., le avisará de los límites de stocks, mínimo y máximo por artículo, etc...

Todo para la llevanza de su almacén.

**14.900 ptas.**

disponibles para: PCW 8256  
PCW 8512

Asimismo, Contabilidad disponible para CPC 664/6128 **9.900 ptas.**

**OFITES INFORMATICA**

**DISTRIBUIDOR EXCLUSIVO DE ORDEMANIA SOFT**

IVA NO INCLUIDO

Condiciones especiales para distribuidores

Si tiene alguna dificultad en obtenerlos dirijase a

Saez Hnos.

12 Quesa

Parre S.A.





# Ofites Informática

## Presenta: la tableta gráfica

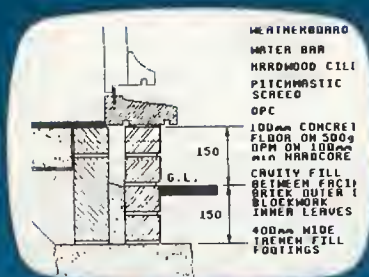
### GRAFPAD II-

#### LO ULTIMO EN DISPOSITIVOS DE ENTRADA DE GRAFICOS PARA AMSTRAD, COMMODORE Y BBC

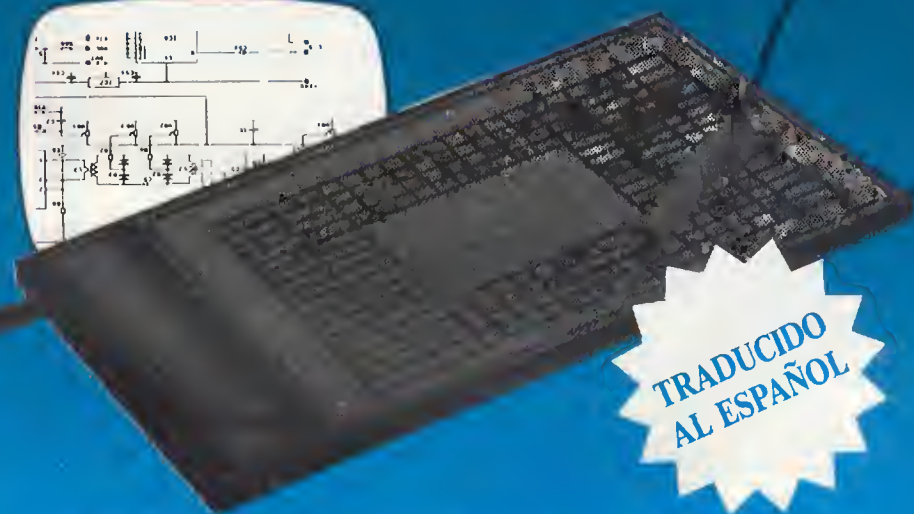
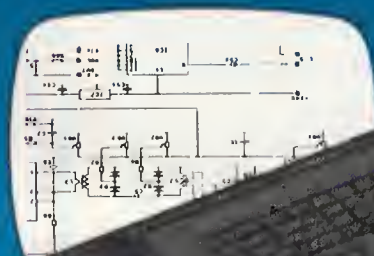
La primera tableta gráfica, de bajo costo, en ofrecer la duración y prestaciones requeridas por las aplicaciones de negocios, industria, hogar y educación. Es pequeña, exacta y segura. No necesita ajustes ni mantenimiento preventivo. GRAFPAD II es un producto único que pone la potencia de la tecnología moderna bajo el control del usuario.



DIBUJO A MANO ALZADA  
SOFTWARE DE ICONOS



DISEÑO DE ARQUITECTURA  
CON SOFTWARE DX



TRADUCIDO  
AL ESPAÑOL

COMBINA EN UN UNICO DISPOSITIVO TODAS LAS PRESTACIONES DE LOS INTENTOS PREVIOS DE MECANISMOS DE ENTRADA DE GRAFICOS. LAS APLICACIONES SON MAS NUMEROSAS QUE EN LOS DEMAS DISPOSITIVOS COMUNES E INCLUYEN:

- selección de opciones
- entrada de modelos
- recogida de datos
- diseño lógico
- diseño de circuitos
- creación de imágenes
- almacenamiento de imágenes
- recuperación de imágenes
- diseño para construcción
- C.A.D. (diseño asistido por ordenador)
- ilustración de textos
- juegos
- diseño de muestras
- educación
- diseño PCB

#### ESPECIFICACIONES

RESOLUCION:

1.280 x 1.024 pixels.

PRECISION:

1 pixel.

TASA DE SALIDA:

2.000 pares de coordenadas por segundo.

INTERFACE:

paralelo.

ORIGEN:

borde superior izquierdo o seleccionable.

DIMENSIONES:

350 x 260 x 12 mm.

DISPONIBLE AMSTRAD:

CASSETTE .... 23 900 ptas

DISCO ..... 25 900 ptas.

(IVA NO INCLUIDO)

- FACIL DE USAR.
- RAZADO PCB.
- C.A.D.
- AREA DE DISEÑO DIN A4.
- COLOR EN ALTA RESOLUCION.
- USO EN HOGAR Y NEGOCIOS.
- VARIEDAD DE PROGRAMAS DISPONIBLES.
- DIBUJO A MANO ALZADA.
- DIAGRAMAS DE CIRCUITOS.

DE VENTA EN LOS MEJORES COMERCIOS DE INFORMATICA  
Si Vd. tiene alguna dificultad para obtener la tableta gráfica, puede dirigirse a:



Avda. Isabel II, 16 -8º

Tel. 455544 - 455533

Télex 36698

21011 SAN SEBASTIAN

CONDICIONES ESPECIALES PARA DISTRIBUIDORES



# DR DRAW

Fc. Javier Barceló T.

**Hace algún tiempo se presentaba en el Banco de pruebas el primer programa de gráficos realizado por Digital Research que se había adaptado a los AMSTRAD 6128 Y 8256, el D.R. Graph. Hoy, de la mano de los mismos autores, esta sección analiza otro programa basado en los gráficos, aunque naturalmente, con un enfoque distinto: El DR. DRAW.**



Con una presentación idéntica y, unas instrucciones igual de completos que el otro programa, aunque también en inglés, este programa va dirigido a un campo distinto. DR. Draw está pensado para realizar dibujo lineal y, por supuesto, para cualquier cosa que a alguien se le ocurra, mientras se pueda realizar a base de circunferencias y líneas rectas.

## Instalación

Además del libro de instrucciones, el programa va acompañado de un pequeño folleto que explica, de una manera muy clara, el procedimiento a seguir para crear una copia del programa lista para utilizar en nuestro ordenador. El proceso es diferente para el 6128 y el 8256, pero en ambas se obtiene un disco con los programas necesarios para funcionar y, que además, dispone de un pequeño espacio libre para guardar algún dibujo. Pero atención a esto. El espacio es pequeño de verdad y, de hecho, en la prueba se grabaron en disco dos dibujos y, al ir a salvar el tercero, el disco estaba lleno. Conviene, pues, disponer al menos de otra cara libre de disco, para no tener problemas de este tipo. El programa hace uso del disco virtual para almacenar algunos programas, con lo que la necesidad de estar dando la vuelta al disco cada dos por tres queda muy omortiguada.

Los usuarios del 6128 están obligados a realizar algún paso más que los del 8256 para obtener esta copia. Esto es debido a que el 8256 viene ya con impresora, por lo que no se necesita indicar las características de la misma al programa, dado que ésta es siempre la misma. Sin embargo, los usuarios del 6128, al poder elegir entre multitud de impresoras diferentes, se ven obligados a definir las características de su impresora al programa creando un fichero **ASSIGN.SYS**, y siguiendo las instrucciones del manual de **CPIM** sobre programas que usen GSX.

## Descripción general

Al cargar el programa, aparece el modo de pantalla que va a ser utilizado en todas ellas.







El primer menú se encarga de las funciones ya habituales de crear, grabar, cargar, editar, imprimir, obtener el directorio del disco y acabar el programa. Por cierto, que la opción de obtener el directorio sólo proporciona los nombres de los programas que se hayan realizado a través de este programa, ignorando los demás que estén en el disco.

Al seleccionar la opción CREAR, pregunta el nombre que se le da al dibujo y, cambia la pantalla apareciendo en la primera línea de la misma el segunda menú de opciones y, en la ventana principal una cuadrícula de puntos que facilitan notablemente la realización de dibujos. Esta cuadrícula se puede cambiar de tamaño, permitiendo así distintos tamaños de letras y dibujos verdaderamente operativos.

Las funciones de cargar y editar también dan paso al segunda menú, funcionando igual que el anterior, excepto que usan un dibujo previamente realizada. Si se ha realizada algún dibujo y se desea cargar otro, el programa pregunta si se quiere grabar el dibujo que está en la memoria, evitando así algún borrado involuntario.

## Banco de PRUEBAS

**Lines.:** Pide dos puntos y, los une con una línea recta.

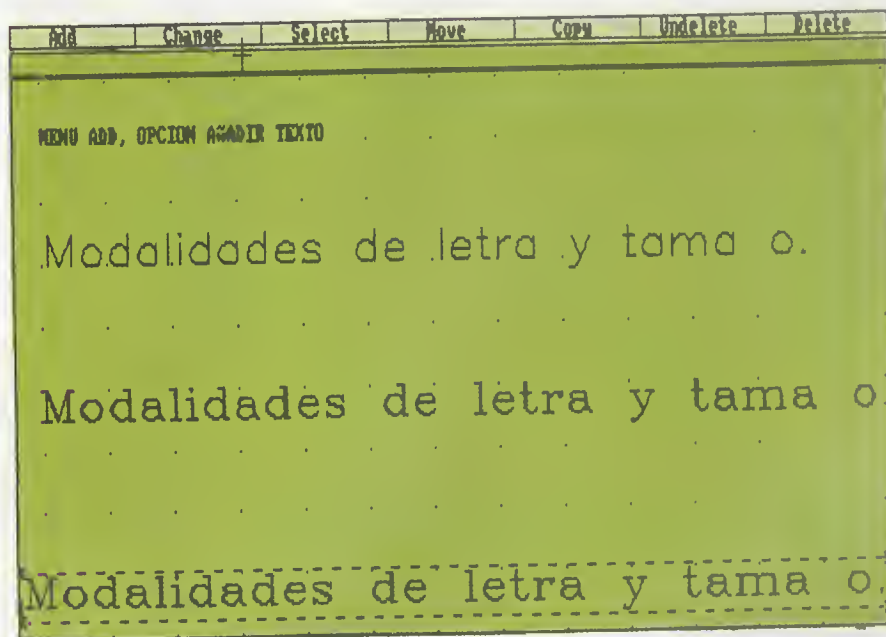
**Markers:** Permite situar puntos en el dibujo, para servirse de ellos como referencia, pero que no aparecen en el dibujo que imprime.

**Bar.:** Crea un rectángulo en base a dar das de los vértices apuestas del mismo.

Además, permite elegir al realizar una de estas figuras si se desea vacía o rellena, en cuyo caso sombrea el contenido de la misma.

La segunda opción de este menú, CHANGE, cambio, permite variar las siguientes parámetros:

**Style.:** Cambia el estilo de la letra, pudiéndose optar entre cuatro tipos, llamadas máquina, simple, compleja e itálica. El único fallo de las tres últimas estilos es que no llevan incorporada la letra ñ.



La primera línea es la línea de opciones, como si fuera un menú y, en la segunda línea pueden salir, bien subopciones de alguna opción elegida, o bien las mensajes que da el programa. El resto de la pantalla, en otra recuadro, es la ventana en la que se hace el dibujo. Pero no por esto tiene que ser ese el tamaño máximo del dibujo, dado que éste se puede correr bien arriba o abajo, bien a izquierda o derecha, y seguir dibujando. No obstante, sólo se imprimirá la parte del dibujo que se encuentre en ese momento en la pantalla, pero como, entre otras facilidades, el dibujo se puede cambiar de tamaño, se puede reducir para que quepa en el papel, aunque se dibuje más grande, por resultar más fácil. La selección de las opciones, así como el dibujo, en realidad todo, se hace a base de mover el cursor por la pantalla con la tecla de cursor, pulsando la tecla espaciadora cuanto éste está sobre la opción elegida y, pulsando RETURN cuando queramos cambiar de opción. Para los que dispongan de ratón o joystick, éste se puede utilizar también.

Pasando ya al menú de la opción CREATE, la primera opción de este menú, ADD, añadir, proporciona en la segunda línea de pantalla otra serie de subopciones:

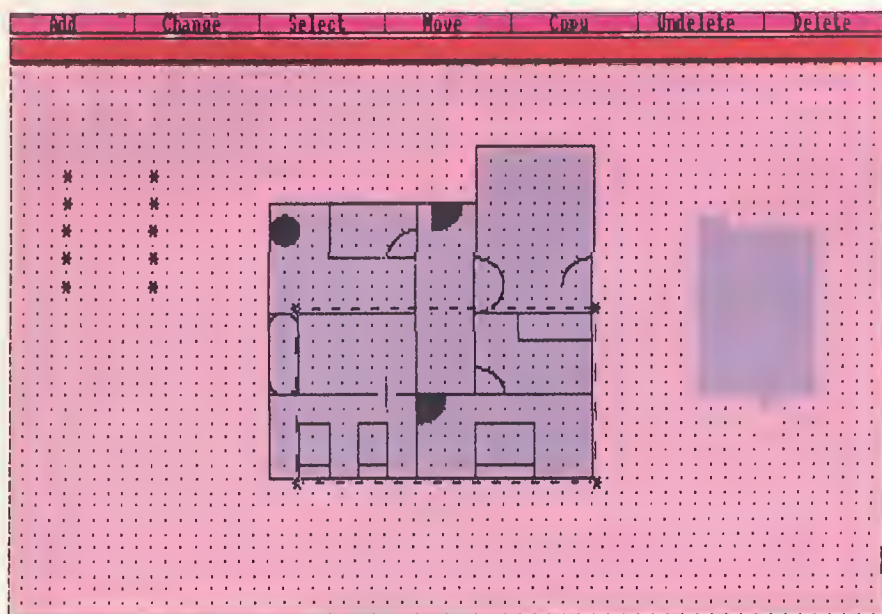
**Text.:** Para incluir textos o intercalarlos en el dibujo.

**Polygon.:** Permite crear una figura geométrica. Se eligen los vértices de la misma y, el programa los une con líneas rectas de forma automática.

**Circle.:** Pide el centro del círculo y su radio, realizándolo también de manera automática.

**Arc.:** Pide el centro del arco, el punto del principio del mismo y, el punto final, dibujándolo en el sentido contrario al de las agujas del reloj.

**View.:** Permite mover la ventana de dibujo por el mismo, en el caso de que éste ocupe más espacio que la pantalla, o desplazar y cambiar de tamaño el dibujo a voluntad. Para esta, dispone de función de Zoom, que permite aumentar o disminuir el tamaño; una función llamada PAN que permite fijar el punto que se convertirá en el nuevo centro de la pantalla, con lo que en realidad se mueve el dibujo, en todos los sentidos. Otra función interesante de este submenú es Put to Back. Esta función tiene su utilidad en el caso de que se estén realizando varias dibujos, si alguna de ellas tapa a otro. En este caso, la opción permite cambiar el elemento que está por delante por el que esté detrás, haciendo visible uno u otro.



**Scale.:** Esta opción permite variar el tamaño de las letras, excepto del tipo máquina. En esta opción hay que señalar un recuadro donde se encuentra el texto a cambiar y, otro recuadro que es en el que el programa «**mete**» el texto en la nueva escala. Se decide pues, el tamaño según el del segundo recuadro. También permite cambiar el tamaño de dibujos, teniendo además dos variantes. En la primera, el dibujo es reducido o aumentado proporcionalmente y, en la segunda, de manera no proporcionada, pudiéndose así obtener algunos efectos muy divertidos.

**Layout.:** Esta es una función muy importante. Dr. Draw sólo imprimir la parte de dibujo que aparece en la pantalla. Por tanto, de nada nos valdría realizar un dibujo mayor, si no existiera esta función. Nos permite seleccionar un recuadro que es el que se va a imprimir, de manera que si se elige un recuadro mayor del tamaño de la pantalla, reduce el dibujo para que éste quepa en ella, o bien, si sólo deseamos imprimir parte de lo representado en la misma, aumenta esta parte, de manera que sea esta parte la que ocupe la pantalla. El submenú tiene hoja DIN A4, que el dibujo se sitúe de forma horizontal, de forma vertical, o bien como sale en la pantalla. Esto permite utilizar mejor la superficie de impresión, adaptándola al dibujo.

**Color:** Para los afortunados poseedores de un **Amstrad CPC 6128** con monitor en color, permite elegir el color en pantalla de los dibujos. Si encima tienen un plotter con colores del dibujo puede realizarse en colores, aunque desgraciadamente no es una posibilidad al alcance de la mayoría...

**Grid.:** La cuadrícula de la que antes se hablaba como puntos de referencia, tiene una importancia vital. El cursor con el que se eligen los puntos de referencia o de principio y final de dibujos se sitúa siempre en el punto más cercano a donde se pulse la tecla espaciadora. Es decir, que no se puede poner el centro de una circunferencia entre dos pun-

tos. Por esta y, para permitir la mayor precisión, se puede cambiar la separación, haciendo la cuadrícula más o menos densa. Para esto, se dispone de una opción que aumenta la separación al doble, otra que la reduce a la mitad y, otra que la elimina totalmente. Al variar el tamaño de la cuadrícula no cambia el del dibujo, pero permite completarlo con más precisión.

**Redraw.:** Esta función es útil si, por ejemplo, hay dos dibujos en la pantalla que tengan partes que se tapen. Si se borra uno de los dos dibujos, la parte que coincide con el otro también se borra. Esta función reconstruye esta parte, redibujándola (*pero no en el dibujo borrado*).

La tercera opción de este menú es **SELECT**. Todos los cambios que se realizan en el tipo, tamaño, copia, borrado, etc... se hacen en la parte de dibujo que se ha realizado la última y, que en la pantalla aparece siempre recuadrado. Si lo que deseamos es cambiar una parte realizada anteriormente, hay que seleccionarla. Esta opción pide dos vértices opuestos de un rectángulo, en el interior del cual debe estar la figura o texto a modificar.

La cuarta opción es **COPY**, que permite copiar cualquier elemento previamente seleccionado, cuantas veces se desee.

La quinta opción **UNDELETE** recupera el último dibujo previamente borrado.

La sexta opción, **DELETE**, borra la porción del dibujo que se encuentre seleccionada en ese momento.

Dentro del menú principal, existen otras opciones que permiten cargar un dibujo previamente grabado en disco (**RECALL**), volver al dibujo que se estaba realizando (**EDIT**), grabar el dibujo en el disco (**SAVE**), obtener un directorio parcial del disco, con los programas de dibujos que se encuentren en esa cara del disco (**DIRECTORY**). Se puede optar por otro disco anteponiendo al nombre del programa una letra distinta de la de la unidad de disco en curso.

La opción **OUTPUT** es la que realiza el dibujo en la impresora o el plotter. Posee además de éstas, dos opciones para definir la salida del dibujo a un monitor en color, y una televisión en blanco y negro. Elegida esta opción, pregunta el nombre del dibujo a imprimir. Esto es así porque imprime un dibujo que esté grabado en disco. Es decir, que antes de imprimir un dibujo, es necesario grabarlo antes.

La opción **EXIT** sale al sistema operativo, dando por finalizado el trabajo.

## Conclusiones

El estilo de Digital Research en la realización de programas tiene una fama muy parecida. Este producto funciona sin ninguna pega y, tiene una presentación impecable. Su libro de instrucciones está en inglés, pero francamente claro, de manera que tampoco haya que saber demasiado para manejar el programa en pocos minutos. Además, el programa está diseñado de manera que resulta muy fácil manejar.

La manera de seleccionar las opciones del menú es quizás la única incómoda o lenta de este programa. No se ha probado con un ratón, por lo que no hay referencias para él, pero desde luego con las teclas de cursor resulta bastante incómodo estar a medio dibujo y, tener que estar subiéndolo al menú y bajándolo al dibujo constantemente.

Las cabeceras y otros tipos de texto pueden resultar francamente bonitas gracias a las posibilidades de texto que posee, pudiéndose presentar dibujos con un gran nivel de perfección.

En cuanto al resto, las posibilidades son francamente buenas. Se pueden realizar dibujos técnicos de todo tipo, siempre que las curvas de los mismos se puedan solventar a base de arcos de circunferencias. Quizás la impresión resulte un poco lenta, pero este es un problema de Hardware y no de programa.

En resumen, este programa servirá para introducir el **Amstrad** en áreas de negocios que hasta ahora estaban reservadas a ordenadores de precios muy superiores.

## FICHA TECNICA

NOMBRE: DR. DRAW  
AUTOR: DIGITAL RESEARCH.  
ORDENADOR MINIMO: CPC 6128 fósforo verde, e impresora.  
ORDENADOR ACONSEJABLE: Cualquier Amstrad con disco, y si se puede con PLOTTER.  
PRECIO: 15.100 ptas. + IVA.  
DISTRIBUIDOR: MICROBYTE.  
C/ San Gerardo, 59. 28035 Madrid. Tel.: (91) 656 50 02.





# Presenta las últimas novedades para AMSTRAD

Con pantallas e instrucciones en CASTELLANO.

## PANZADROME

*¡Nada podrá parar a este tanque!*



En el juego PANZADROME verás una isla habitada por tanques robots de varios niveles de peligrosidad. Tu misión es aniquilarlos. Para ello dispones de un tanque con un buscador y un lanzador de minas. Podrás conseguir morteros buscándolos por toda la isla, podrás reparar y verificar el estado del blindaje, pero estate atento: tienes combustible limitado.

## TOAD RUNNER

*¿Me liberaré de la Maldición?*

Casi he olvidado mi propia imagen... Yo era el Príncipe... Aquella noche estará conmigo para siempre. La luna, la locura en el aire y aquella diabólica maldición por la que fui convertido en el ¡TOAD RUNNER! (Sapo Corredor).



Si alguna vez he de recuperar mi majestuosa apariencia, habré de encontrar a mi princesa y si no tengo éxito... no podré decir más que aún soy el ¡TOAD RUNNER!.



Editado por DRO SOFT. Fundadores, 3 - 28028 Madrid  
Tlfs.: 255 45 00 / 09





# TIRO CON ARCO

*La gente que se dedica a hacer programas no tiene una edad definida; los hay de todos los «rangos» posibles. Normalmente se piensa que corta edad es sinónimo de programas simples y no demasiado cuidados, cosa que, afortunadamente, no es cierto.*



Como prueba, aquí va esta creación de uno de nuestros lectores de 13 años de edad, que está realizada con bastante cuidado y demuestra unos conocimientos de programación considerables. Llama la atención, además, la forma de usarlo, extremadamente sencilla y elegante. Creo que os vais a divertir, y hasta aprender un poco, del «Tiro con arco».

Se elige nivel de 1 al 4, que cambian la velocidad de la diana. El nivel 4 elige la velocidad al azar.

En una de las ventanas marca las flechas restantes, en otra los grados del tiro, en otra el récord y los puntos, y en otra los puntos conseguidos por el último tiro.

Para tirar y sumar grados hay que pulsar (espacio). La flecha tiene que dar en la diana que pasa de arriba a abajo. Luego se marcará en la diana vista de frente y dará los puntos.

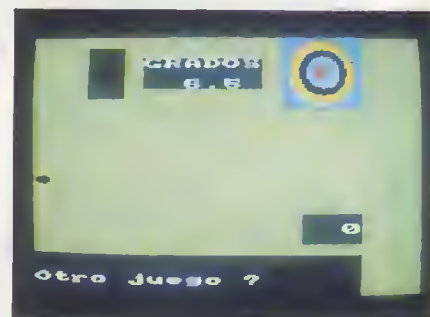
Camilo Alvarado

## SUBROUTINAS

60-100	Elige nivel
110-340	Subrutina principal
350-560	Localiza blanco y da puntuación
570-660	Otro juego
670-790	Inicializa variables, tintas y ventanas
800-880	Dibuja diana
890-1100	Redefine caracteres
1110-1140	Dibuja flechas iniciales
1150-1180	Borra flechas de una en una

## VARIABLES

IS, AS	Tecla pulsada
I	Nivel elegido
GR	Grados del tiro
X, Y	Coordenadas de la flecha
J	Posición de la diana
FL	Flechas
BL	Posición de la flecha en la diana
XX, YY	Coordenadas del blanco
XX1, YY1	Coordenadas del blanco anterior
SC, P	Puntos
HI	Récord
HOM\$	Tirador
FL\$	Flecha





```

10 GOSUB 1190
20 MODE 0:CLG 13
30 GOSUB 670
40 GOSUB 890
50 GOSUB 800
60
70 CLS#6:PEN #6,12:LOCATE #6,1,1:PR
INT #6,"Nivel (1-4)"
80 i$=INKEY$:i=VAL(i$)
90 IF i<1 OR i>4 THEN GOTO 80 ELSE
PEN #6,7:LOCATE #6,5,2:PRINT #6,i$
FOR r=1 TO 1000:NEXT:CLS#6
100 GOSUB 1110
110
120 f1=f1-1:WHILE f1>1
130 FOR rr=1 TO 500:NEXT
140 GOSUB 1150
150 IF i=4 THEN i=INT(RND*2+1) EL
E i=i
160 r=(2*i+1)+4:gr=0:x=3:y=14:j=400
170 WHILE INKEY$<>" ":FOR rr=1 TO 2
00:NEXT:GOSUB 300:WEND:SOUND 1,220,
50,15,1,1
180 WHILE x<20
190 PEN #4,12:FOR v=1 TO 2
200 LOCATE #4,1,1:PRINT #4,"GRADOS"
;
210 IF INKEY$=" " THEN gr=gr+0.5
220 IF gr>10 THEN gr=10
230 LOCATE #4,2,3:PRINT #4,USING "
#.##";gr:NEXT
240 LOCATE x,y:PRINT " ":LOCATE x+1
,y:PRINT f1$
250 GOSUB 300
260 IF x=18 AND (j>192 AND j<266) T
HEN GOTO 350
270 x=x+1:WEND
280 f1=f1-1:CLS#7:LOCATE x,y:PRINT
" ":WEND
290 GOTO 570
300
310 TAG:MOVE 608,j:PRINT " ";
320 PLOT -3,-3,3:j=j-r:FOR jj=j TO
j-64 STEP -16:MOVE 608,jj:PRINT d$;
iNEXT:TAGOFF
330 IF j=64 THEN CLS#7:LOCATE x,y:P
RINT " ":GOTO 160
340 RETURN
350
360 b1=j-192
370 SOUND 1,500,30,15
380 IF b1>60 OR b1<5 THEN p=0
390 IF (b1<61 AND b1>54) OR (b1>5 A
ND b1<12) THEN p=50
400 IF (b1<54 AND b1>47) OR (b1>12
AND b1<19) THEN p=60
410 IF (b1<47 AND b1>38) OR (b1>19
AND b1<28) THEN p=80

```



```

420 IF (b1<38 AND b1>28) THEN p=100
430 IF (b1<33 OR b1>32) AND gr=4.5
THEN p=110
440 IF gr>5 THEN p2=(10-gr)/2 ELSE
p2=gr/2
450 p1=(p/80)*p2
460 IF gr>9.5 OR gr<0.5 THEN p1=0
470 xx=(b1*1.56)+432:yy=(gr*10)+293
480 PLOT xx1,yy1,15:PLOT xx1+2,yy1:
PLOT xx1,yy1+2:PLOT xx1+2,yy1+2
490 PLOT xx,yy,14:PLOT xx+2,yy:PLOT
xx,yy+2:PLOT xx+2,yy+2
500 xx1=xx:yy1=yy
510 sc=sc+INT(p*p1)
520 PEN #2,12:LOCATE #2,1,2:PRINT #
2,USING "###";INT(p*p1);
530 PEN #6,7:LOCATE #6,1,1:PRINT #6
,"Hi Sc":LOCATE #6,12,1:PRINT #6,"S
core"
540 PEN #6,12:LOCATE #6,1,3:PRINT #
6,USING "####";hi:LOCATE #6,13,3:PR
INT #6,USING "####";sc
550 FOR r=1 TO 1000:NEXT

```



```

560 CLS#7:GOTO 120
570
580 CLS #6:PEN #6,12
590 LOCATE #6,1,3:PRINT #6,"Otro ju
ego ?"
600 a$=INKEY$
610 IF a$="S" OR a$="s" THEN GOTO 6
40
620 IF a$="N" OR a$="n" THEN GOTO 6
60
630 GOTO 600
640 CLS #6:IF sc>hi THEN hi=sc
650 CLS#5:xx1=-4:yy1=-4:sc=0:f1=12:
GOTO 50
660 MODE 1:PAPER 0:PEN 1:BORDER 1:C
LS:END
670
680 sc=0:hi=0:f1=12:xx1=-4:yy1=-4:E
NV 1,14,1,1,10,-1,30:ENT 1,100,1,1,
100,-1,10
690 INK 3,13:INK 4,15:INK 5,0:INK 6
,14:INK 7,6:INK 8,0:INK 9,15:INK 12
,26:INK 13,18:INK 14,26,3:INK 15,26
700
710 BORDER 0
720 WINDOW #0,1,20,1,25:PAPER #0,13
:CLS#0
730 WINDOW #6,1,17,22,25:PAPER #6,8
:CLS #6
740 WINDOW #2,15,17,18,20:PAPER #2,
8:CLS #2
750 WINDOW #3,4,5,2,6:PAPER #3,8:C
LS #3
760 WINDOW #4,7,12,3,5:PAPER #4,8:C
LS #4
770 WINDOW #5,14,17,1,7:PAPER #5,3:
CLS #5
780 WINDOW #7,19,20,1,25:PAPER #7,1
3:CLS#7
790 RETURN
800
810 x=480:y=345:r=50:c=4:GOSUB 850
820 x=481:y=345:r=35:c=5:GOSUB 850
830 x=482:y=345:r=25:c=6:GOSUB 850
840 x=478:y=345:r=10:c=7:GOSUB 850
850 FOR q=y-r TO y+r STEP 2
860 inc=INT(SQR((r*r)-(y-q)*(y-q))
)+0.5)
870 MOVE x-inc,q:DRAW x+inc,q,c
880 NEXT:RETURN

```

```

890
900 SYMBOL AFTER 230
910 SYMBOL 231,0,0,0,0,60,126,126,1
26
920 SYMBOL 232,126,60,0,0,0,0,0,0
930 SYMBOL 233,0,0,0,31,3,0,0,0
940 SYMBOL 234,0,0,0,0,240,248,8
950 SYMBOL 235,0,0,0,0,0,63,63
960 SYMBOL 236,0,0,24,31,31,0,0,0
970 SYMBOL 237,192,192,192,192,192,
0,0,0
980 SYMBOL 238,0,0,0,96,62,96,0,0
990 SYMBOL 240,192,192,192,192,192,
192,192,192
1000
1010 z=29:GOSUB 1050:hom$=a$
1020 LOCATE 1,14:PRINT hom$
1030 f1$=CHR$(238)
1040 d$=CHR$(240)
1050 RESTORE 1100
1060 a$=""
1070 FOR n=1 TO z
1080 READ nu:a$=a$+CHR$(nu)
1090 NEXT:RETURN
1100 DATA 22,1,15,8,231,15,9,234,8,
15,8,10,,8,232,15,9,237,8,8,236,11,
8,233,15,3,235,22,0
1110 FOR f=1 TO 5
1120 LOCATE #3,1,f:PEN #3,3:PRINT #
3,f1$;f1$;
1130 NEXT
1140 RETURN
1150
1160 h=(f1 MOD 2)+1:g=f1\2
1170 LOCATE #3,h,g:PRINT #3," ";SO
UNO 1,50,30,15
1180 RETURN
1190
1200 INK 0,0:INK 1,24:INK 2,0:INK 3
,6:MODE 1:PAPER 0:CLS:BORDER 0
1210 LOCATE 1,25:PEN 2:PRINT "Super
ARCO"
1220 FOR x=0 TO 160 STEP 2
1230 FOR y=0 TO 16 STEP 2
1240 IF TEST(x,y)=2 THEN PLOT x*2+1
60,y*2+350,1:PLOT x*2+162,y*2+352:P
LOT x*2+160,y*2+352:PLOT x*2+162,y*
2+350
1250 NEXT y,x
1260 PEN 3:LOCATE 10,12:PRINT "Dese
a instrucciones ?"
1270 a$=INKEY$
1280 IF a$="S" OR a$="s" THEN GOTO
1310
1290 IF a$="N" OR a$="n" THEN RETUR
N
1300 GOTO 1270
1310 WINDOW #1,1,40,1,4:WINDOW #0,1
,40,7,25:CLS #0
1320 LOCATE 1,1:PRINT "Debes tirar
a la diana que pasa de ":LOCATE 1,3
:PRINT "arriba a abajo."
1330 LOCATE 1,5:PRINT "Ten en cuent
a los grados."
1340 LOCATE 1,7:PRINT "Utiliza (EPA
C10) para tirar e ir ":LOCATE 1,9:P
RINT "sumando grados"
1350 LOCATE 10,12:PEN 1:PRINT "Puls
a una tecla"
1360 CALL &B818:RETURN

```



**P**ara que tus dedos  
 no realicen el trabajo duro, M.H. AMS  
 TRAD lo hace por ti. Todos los listados que incluyen  
 este logotipo se encuentran a tu disposición en un cas-  
 sette mensual, solicítalo



# Mercado común

Con el objeto de fomentar las relaciones entre las usuarias de AMSTRAD, **MERCADO COMUN** te ofrece sus páginas para publicar las pequeñas anuncios que relacionadas con el ordenador y su mundo se ajusten al formato indicada a continuación.

En **MERCADO COMUN** tienen cabida, anuncios de ventas, compras, clubs de usuarias de AMSTRAD, programadores, y en general cualquier clase de anuncio que pueda servir de utilidad a nuestras lectoras.

Envíanas tu anuncio mecanografiado a: **HOBBY PRESS, S.A.**

**AMSTRAD SEMANAL.**

Apartada de correos 54.062  
28080 MADRID

**¡ABSTENERSE PIRATAS!**

**Vendo** ratón AMX MOUSE para **Amstrad** CPC-464 comprado en Semana Santa. Precio: 12.500, Alexis Gutiérrez. C/ Gutiérrez Rada, 2. Laredo (Cantabria). Tel. (942) 60 62 25.

**Desearía** contactar con usuarios de **Amstrad** 6128 para intercambiar juegos e ideas. Únicamente de Madrid. Llamar de 11 de la mañana a 6 de la tarde al Tel. 845 68 91 o escribir a C/ Fuente Santa, 9 B-D de Colmenar Viejo. Preguntar por Javi, responderé todas las cartas.

**Desearía** contactar con usuarios de **Amstrad** PCW-8256 para intercambio de ideas, información y programas. Preferentemente sobre el área de programación en BASIC y COBOL. También cambio fotocopias de un libro de Cobol. (ed. Paraninfo) por las de uno de CP/M. Tel. (91) 718 49 49 (Madrid). Preguntar por José.

**Vendo Amstrad** 664, monitor fósforo verde con pantalla reflectante, 25 programas profesionales. Micropen, Stock, contabilidad, Micrascript, etc. 30 juegos: Manic-Miner, Superpipeline, Cdename-Mat, etc. Utilidades: Tascopy, Tosprint, Music-Maestro, etc. Todo por 95.000 pts. También vendo impresora SEIKOSHA SP-800, doble tracción-fricción, 96 c.p.s. 9 tipos de letras, centronic, por 57.000 pts. Isabel. Madrid, Tel. (91) 719 22 31.

**Vendo** CPC-464, en perfecto estado, oferta interesante para programadores (código máquina, ensamblador, revistas y libros sobre el ordenador), interesados llamar al Tel. 460 45 83 de 3,30 a 5,00 de la tarde preguntar por Javier.

**Vendo Amstrad** CPC-464 monitor color comprado en junio del 85 con garantía, 2 manuales, todos los números de micro-hobby **Amstrad** y **Amstrad** USER y 30 programas en cinta 65.000 pts. Tel. (93) 211 51 79 noches.

**Vendo** con urgencia: **Amstrad** 6128, Impresora Admate 100, cassette para ordenador, colección completa de la revista **Amstrad** Semanal. Software de utilidades: 2 discos maestros, Mastercalc, Amsword II, Microscript, Micropen, Microspread, Zedis II, Oddjob, Sistem X, Devpac, Planificador de Proyectos, Toma de decisiones. Software de entretenimiento: Star Watcher, 3D-Voice Chess. En cinta: Decathlon, ¡Oh Mummy!, El laberinto del Sultán, Almirante Graf Spee, Mensaje de Andrómeda, Roland in Time, contabilidad doméstica, Knight Lore, Star Comando, Battle for Midway, Master Chess, Amsbase, Southern Belle, Amsgolf, Condename Mat..., y más programas en disco y cinta. El increíble precio de esta oferta es de 120.000 pts. Contactar con el Tel. 457 29 43. Preguntar por Isabel.

**Desearía** contactar con usuarios de toda España, para intercambio de programas ideas y demás. Los interesados se pueden dirigir a: Francisco Espejo Ramírez. Urb. «El Retortillo» portal 4-1.º B. Chiclana de la Frontera. 11000 (Cádiz).

**Vendo** CPC-464 prácticamente sin uso completo 70.000 pts. Alberto González. Tel. 274 89 90. (Madrid).

**Desearía** intercambiar los siguientes juegos: Skyfox, Knight Lore, Raid Over Moscov, Hunter Killer, Astro Attack, Roland in the Caves. Por los siguiente: Dun Darach, One on One, Hacker, Decathlon. No es necesario tener todos los pedidos. Escribir a Juan Pérez. C/ Zapatería, 47. Alcántara (Cáceres).

**Cambio** fotocopias de los manuales de AMSCALC (ver. 5), PASCAL y Gena-Mona de HISOFT (juntos o por separado) por 25 programas comerciales a elegir. Escribir a Fermín García Nieto. C/ Santa Catalina, 3, 1.º A. 30004 Murcia.

**Soy** usuario de un **Amstrad** CPC-464, y estoy interesado en intercambiar programas de todo tipo (en especial juegos) sólo comerciales. Interesados llamar al Tel. (956) 66 72 47 o bien escribir a, Miguel A. Moreno Aparicio. Urb. Bahía de Algeciras, Bl. 15-3.º C.

**Desearía** contactar con usuarios de **Amstrad** (disco o cinta) para intercambio de juegos. También vendo el siguiente lote de programas en cinta al precio de 2.300 pts.: Simulador de vuelo 737, Ghoul, Airwolf, Chickie, Pimbal Wizard, Killer Gorila, Southern Belle y Spannerman, o también lo vendo a 300 pts. cada uno. Jesús. Tel. (91) 845 39 80.



Version  
SPECTRUM,  
AMSTRAD Y COMMODORE

# Juegos ESTRATEGIA

## Alistate a LA BATALLA DE INGLATERRA ha comenzado

Pilota tu «Spitfire»  
sobre el Canal de La Mancha  
y defiende el  
destino  
del mundo libre.



Oferta especial  
hasta el 31  
de noviembre:  
PIDE TRES NUMEROS  
Y PAGA  
SOLO DOS.



ENVIE HOY MISMO ESTE CUPON AL APARTADO 232 DE ALCOBENDAS (Madrid)

- ☐ Deseo recibir en mi domicilio tres ejemplares de **Juegos & Estrategia** al precio especial de 2.255 ptas., lo que me supone adquirir tres y pagar sólo dos. Marco los tres ejemplares que deseo con una cruz.
- ☐ Deseo recibir un solo ejemplar de **Juegos & Estrategia** al precio de 1.125 ptas. Marco con una cruz el ejemplar que desea recibir.

### Spectrum

- N.º 1 ☐ Arnhem  
N.º 2 ☐ Ratas del Desierto  
N.º 3 ☐ OTAN Alerta  
War Zone

Especial 1 ☐ Elecciones Generales

N.º 4 ☐ Su mejor hora (La batalla de Inglaterra)

### Amstrad

- ☐ Arnhem  
☐ Ratas del Desierto  
☐ Teatro de Europa  
War Zone

☐ La batalla de Inglaterra

### Commodore

☐ Teatro de Europa

☐ La batalla de Inglaterra

NOMBRE \_\_\_\_\_

DIRECCION \_\_\_\_\_

LOCALIDAD \_\_\_\_\_

C. POSTAL \_\_\_\_\_

TELEFONO \_\_\_\_\_

PROVINCIA \_\_\_\_\_

PROFESION \_\_\_\_\_

Fecha de  
nacimiento

Formo de pago:

☐ Talón bancario a nombre de Hobby Press, S.A. ☐ Giro Postal a nombre de Hobby Press, S.A., n.º de giro

☐ Tarjeta de crédito: Visa n.º \_\_\_\_\_

Master Charge n.º \_\_\_\_\_

American Express n.º \_\_\_\_\_

Fecha de caducidad de la tarjeta \_\_\_\_\_

Fecha y firma \_\_\_\_\_



# RASTREO DE LOS PROCEDIMIENTOS RECURSIVOS

*En esta ocasión pretendemos, mediante un método esquemático, hacer un seguimiento de los procedimientos LOGO basados en la recursividad. Con ello se intenta poner fin a esta cuestión, una de las más importantes de éste y otros lenguajes de programación.*



Recordemos, lo hemos tratado en anteriores artículos, que necesitamos de la RECURSIVIDAD para crear procedimientos en los que ciertos pasos se repiten, obedeciendo a los requerimientos de nuestro problema. El LOGO de los GRAFICOS DE LA TORTUGA, así como el ARITMETICO, han sido muestras de ello.

Se verá más adelante esta misma necesidad, cuando nos introduzcamos en el tema de tratamiento de PALABRAS Y LISTAS.

Cuando se llega a dominar el arte de la programación en un determinado lenguaje de ALTO NIVEL, cuestión no tan fácil quizá como algunos desearíamos, se observa de manera agradablemente sorprendente que, paso a paso, las diversas partes de nuestro programa, en su ejecución, van cumpliendo los objetivos que a priori les hemos encomendado.

En alguna ocasión cometemos pequeñas equivocaciones que serán rectificadas tan pronto como podamos averiguar su causa. Posiblemente, para ello, hayamos necesitado proceder a la EJECUCION del programa.

Por último, se sabe que en ciertas ocasiones, planteamos un sector de programa previendo un determinado efecto que al ejecutarlo no se produce. La razón del imprevisto efecto no acabo de explicarse. Se hace imprescindible «reestudiar» el significado y alcance de alguna INSTRUCCION y/o PRIMITIVA.

Cuando esto se ha hecho, se procede igualmente con un conjunto de instrucciones y/o primitivas, es decir, BLOQUES o SUBPROGRAMAS y/o PROCEDIMIENTOS. A estas fases del trabajo de programación podríamos bautizarlas con el nombre de RASTREO.

Es lo que a continuación haremos con algunos procedimientos cuyo estudio nos va a servir para completar el tema de la RECURSIVIDAD.

Para ello presentamos en primer lugar, éste al que hemos llamado **cuenta**.

## Procedimiento cuenta

```
to cuenta :num
if :num=0 [stop]
pr :num
cuenta :num - 1
end
```

Como ya sabemos editarlo y ejecutarlo, lo hacemos. Como resultado obtenemos algo esperado y previsto, no sólo por el nombre que le hemos dado, sino también porque nuestra mente ya se va acostumbrando. He lo aquí.

```
?cuenta 3
3
2
1
```

Y a continuación presentamos este otro, al que llamaremos **mosqueo**.

## Procedimiento mosqueo

```
to mosqueo :num
if :num=0 [stop]
mosqueo :num - 1
pr :num
end
```

Ejecutémolo. El resultado es lo que a continuación se ve.

```
?mosqueo 3
1
2
3
```

Su nombre está justificado. ¿No es cierto?

Sin embargo, si a ambos se les hubiese dado el mismo nombre, podríamos haber considerado al segundo, como un lapsus. Realmente, la diferencia se limita a la permutación de las líneas tercera y cuarta. Este «error» entre comillas, yo que es intencionado, nos invita e induce a pensar, como otros errores sin comillas. De ello aprendemos.

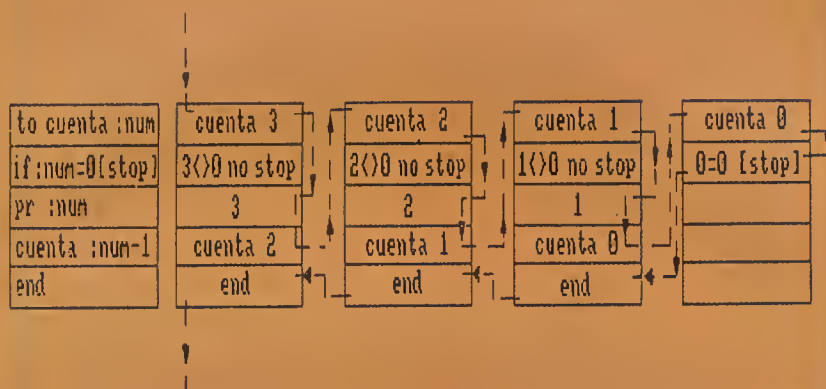
## El método del rastreo

A continuación se presentan lo que algún autor llama «TRAZAS» de ambos procedimientos, y que no es otra cosa que una representación semigráfica del desarrollo de éstos. De esta forma realizaremos el RASTREO. He aquí lo primera.

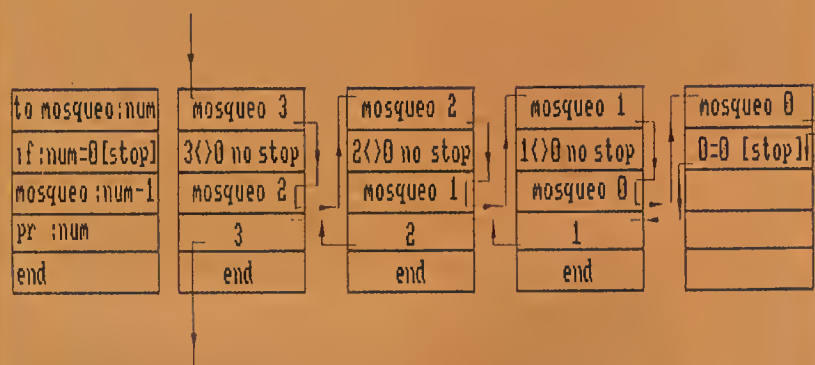


# ANALOGO

## TRAZA DEL PROCEDIMIENTO "cuenta"



## TRAZA DEL PROCEDIMIENTO "mosqueo"



Como aclaración para la interpretación de la traza, señalaremos que la ejecución del procedimiento inicial, el situado a la izquierda, activa el situado a su derecha, y así de forma repetida hasta llegar al último de la derecha. Estos procedimientos se diferencian en el parámetro de entrada solamente. La trayectoria de derecha a izquierda, de regreso, se explica después.

((ver traza del procedimiento «cuenta» ))

Lo que en ella se observa es previsible hasta cierto punto. Realmente, la trayectoria en el sentido de izquierda a derecha, tiene explicación lógica.

La trayectoria que llamaríamos «de regreso», y que se efectúa de derecha a izquierda, va a precisar de algún comentario. Con un poco de detenimiento, será fácil darnos cuenta que su inicio se produce a partir de la situación en la cual, el procedimiento entra en la línea segunda. Hecho que se verifica cuando la condición en ella expresada, se cumple.

A partir de ese momento, la interpretación de esta trayectoria la haremos en base a la que denominaremos si se nos permite, REGLA BASICA DE LA RECURSIVIDAD. Abusando de este permiso nos atreveremos incluso a emitir un enunciado:

CUANDO SE ACTIVA UN PROCEDIMIENTO, EL PROCEDIMIENTO QUE LO ACTIVO QUEDA A LA ESPERA DE QUE EL ACTIVADO ENCUENTRE EN SU EJECUCION UN «STOP». A PARTIR DE ENTONCES, EL ACTIVADOR CONTINUA EL PROCESO, DESPUES DE LA LINEA QUE PRODUJO LA ACTIVACION. EL PROCESO CONTINUA EN EL SENTIDO ACTIVADO-ACTIVADOR, ES DECIR, DE DERECHA A IZQUIERDA.

Al escribir este último párrafo se tiene la misma sensación, creemos, que la percibida por el lector. Ojalá nos equivaquemos. Admitimos que puede resultar farragosa.

Como lo que se pretende es hacer el tema la más clara posible, remitámonos a la TRAZA y volvamos sobre nuestros pasos.

En el procedimiento llamada CUENTA :NUMERO, cuando posamos a su ejecución con el mandato «cuenta :3», el activador es éste y el activado, el denominado «cuenta :2». Cuando por fin es activado «cuenta :0», se cumple la condición expresada por «if». A partir de este momento, el proceso continúa en el sentido «ACTIVADO-ACTIVADOR», o sea de derecha a izquierda, y a partir de la siguiente línea a la activadora, que en nuestra casa es, en todo el proceso, la cuarta empezando por arriba.

Sentimos que todo lo anterior sea arduo. Lo consideramos a pesar de ello, necesario. Quizás este tema sea el más duro del lenguaje, pero su potencia y otras características positivas del mismo, están basadas en él. Otros lenguajes de programación como el «PASCAL», tienen esta característica de la RECURSION. El tema que estamos tocando, en última instancia, y en el peor de los casos serviría como introductorio, e incluso como «aparato de gimnasio», (mental, claro).

La traza del procedimiento «mosqueo» es la que a continuación se presenta.

((ver traza del procedimiento «mosqueo»))

De acuerdo con lo observado al seguir el recorrido según las flechas, se comprende que el orden de aparición de los números tiene que ser el que es, y no otro. Pero para llegar aquí hemos necesitado estudiar la cuestión con un detenimiento mayor que si hubiésemos obtenido un resultado más acorde con lo que en principio, y de forma equivocada, habríamos previsto.

Existen procedimientos más rebuscados que utilizan la RECURSION, cuyo RASTREO resulta más difícil que las presentadas en este trabajo. Para lo que necesitaremos en lo sucesivo, nos detendremos en la expuesta.

SOMOS MAYORISTAS

# MICRO-1

PRECIOS  
INCLUIDO IVA

C/ Duque de Sesto, 50. 28009 Madrid  
Tel.: (91) 275 96 16 - 274 75 02

(Metro O'Donnell o Goya)  
Aparcamiento gratuito en Felipe II

Ofertas en software: 2 programas al precio de 1 y además regalo un reloj digital completamente gratis. ¡¡Asombroso!! ¿Verdad?

BAT MAN	2.300 ptas.	KUNG-FU MASTER	2.300 ptas.
ROCK'N LUCHA	2.300 ptas.	SABOTEUR	2.300 ptas.
YIER AR KUNG FU	2.300 ptas.	PING-PONG	2.300 ptas.
THE WAY OF THE TIGER	2.300 ptas.	MILLION II	2.500 ptas.
WEST BANK	2.100 ptas.	OLE TORO	2.300 ptas.
CAMELOT WARRIORS	2.300 ptas.	TURBO ESPRIT	2.100 ptas.
RAMBO	2.300 ptas.	SABRE WULF	1.650 ptas.
WORLD CUP (DISCO)	3.300 ptas.	BATALLA DE LOS PLANETAS	2.100 ptas.
MILLION II (DISCO)	3.300 ptas.	SABOTEUR-COMBAT LINX DISCO	3.300 ptas.
RAMBO-MATCH DAY (DISCO) 3.300 ptas.			

SOFTWARE DE REGALO (OFERTA 2 x 1)  
DECATHLON BEACH HEAD SOUTHERN BELLE  
DRAGONTORC

LAPIZ OPTICO  
3.295 PTAS.

CASSETTE ESPECIAL ORDENADOR  
4.495 PTAS.

SINTETIZADOR DE VOZ EN  
CASTELLANO  
7.650 PTAS.

AMPLIACION DE MEMORIA ANTA 64 K.3  
12.500 PTAS.

**IMPRESORAS**  
**20 % DE DESCUENTO SOBRE P.V.P.**

TAPA DE METACRILATO 464	895
CABLE CENTRONICS	3.175
CABLE SEGUNDA UNIDAD D.	1.790
CABLE SEPARADORES 6128	1.975
INTERFACE RS232	9.265
CINTA VIRGEN C15	69

CABLE AUDIO	795
CABLE ADAPTADOR 2 JOYSTICK	2.390
CABLE SEPARADORES 464	1.390
CABLE SEPARADOR 8256	2.900
CABLE RS232	2.500
DISKETTES 3"	830

**PRECIOS EXCEPCIONALES PARA TU AMSTRAD  
CPC-464, CPC-6128, PCW-8256**

RATON DE PANTALLA CON SOFTWARE 8.900 PTAS.

**OFERTAS EN JOYSTICKS**

QUICK SHOT I	1.395
QUICK SHOT II	1.695
QUICK SHOT V	1.695

**PEDIDOS CONTRA REEMBOLSO SIN NINGUN GASTO DE ENVIO. TEL. (91) 275 96 16 - 274 75 02  
O ESCRIBIENDO A: MICRO-1. C/ DUQUE DE SESTO, 50. 28009 MADRID.**

Tiendas y distribuidores grandes descuentos.  
Dirigirse a Diproimsa. C/ Galatea, 25. Tel. (91) 274 75 03



# LOS MEJORES PROGRAMAS PROFESIONALES DEL MUNDO

*¡a precios "AMSTRAD"!*

PARA AMSTRAD PCW 8256 Y AMSTRAD CPC 6128

## MICROSOFT

### MULTIPLAN

Uno de los más prestigiosos y completos "hojas de cálculo" del mundo. Rápido y versátil, ofrece prestaciones, como la de relacionar varios hojas entre sí, que no son frecuentes. La capacidad de ejecutar ordenaciones alfabéticas o numéricas, sus posibilidades en cuanto a formato en pantalla y en impresora, los menús en pantalla y la potencia de cálculo, son características distintivas y destacables de MULTIPLAN.

**PVP: 15.100.- Ptas. (+ IVA)**

### MBASIC INTERPRETER

Reconocido como el estándar mundial de los lenguajes intérpretes para microordenadores. Fácil de aprender y utilizar.

**PVP: 15.100.- Ptas. (+ IVA)**

### MBASIC COMPILER

Totalmente compatible con el MBASIC Interpreter pero con una velocidad de ejecución de 3 o 10 veces más rápida. Traduce el código fuente o código objeto y permite una utilización más eficaz del espacio.

**PVP: 15.100.- Ptas. (+ IVA)**

### MS COBOL COMPILER

Lenguaje COBOL según el estándar ANSI, especialmente útil para manejar grandes volúmenes de datos.

**PVP: 48.500.- Ptas. (+ IVA)**

### MS SORT

Flexible programa de ordenación según la técnica de la inserción binaria, utilizable independientemente o incluido en programas escritos en MS COBOL.

**PVP: 15.100.- Ptas. (+ IVA)**

### MS-FORTRAN COMPILER

El lenguaje más utilizado en aplicaciones científicas y de ingeniería, es una patente implementación del ANSI-FORTRAN X3.9

**PVP: 24.900.- Ptas. (+ IVA)**

### MS-MACRO

Un completo paquete de desarrollo que incluye: MS-MACRO ASSEMBLER; MS-LINK, MS-LIB, MS-CREF y DEBUG.

**PVP: 12.000.- Ptas. (+ IVA)**

## ASHTON-TATE

### dBASE II

El Generador de Programas por excelencia. Permite crear bases de datos relacionados a partir de comandos sencillos y sin requerir conocimientos de programación. Las aplicaciones de dBASE II son incontables y cada usuario puede desarrollar los que mejor se adapten a sus necesidades: ficheros y mailings, contabilidades, nóminas, control de costos, control de almacén, facturación, etc. Ampliamente acreditado como uno de los programas más útiles y recomendables de cuantos existen para microordenadores. *Manual en castellano.*

**PVP: 17.800.- Ptas. (+ IVA)**

### DR. DRAW

Programa interactivo para la creación y edición de gráficos y diagramas. Tres elementos básicos —líneas, texto y símbolos— son utilizados para producir gráficos de alta calidad... logos, diagramas de bloques, diagramas de flujo, etc. Los símbolos, tipos de letra y estilos de líneas, pueden alterarse y modificarse a voluntad del usuario.

**PVP: 15.100.- Ptas. (+ IVA)**

### DR. GRAPH

Generador de gráficos —de líneas, barras, columnas y de pastel— de muy sencillo manejo. Permite incluir textos y leyendas con gran flexibilidad de creación y edición.

**PVP: 15.100.- Ptas. (+ IVA)**

### PASCAL MT+

El más rápido PASCAL existente con implementación completa del estándar ISO. Un compilador de código nativo que genera en formato reubicable para usar con su montador de enlaces (linker).

**PVP: 15.100.- Ptas. (+ IVA)**

### CBASIC COMPILER

Versión mejorada del clásico lenguaje CBASIC, con mayor velocidad de ejecución y altamente flexible diseñado especialmente para el desarrollo de programas de gestión. Incluye el linker LK-80, que cambia la salida del compilador con las rutinas de biblioteca y permite el encadenamiento de módulos.

**PVP: 15.100.- Ptas. (+ IVA)**

## micro bit

P.º CASTELLANA, 179-1.º - 28046 MADRID  
Telf. 442 54 33/44





# STAINLESS STEEL



Toda la acción del mundo en este nuevo juego de MIKRO - GEN en el que el protagonista RICKY STEEL a bordo de un coche que lo mismo vuela que navega por un río y que es capaz de disparar sus láser en 12 direcciones distintas o arrojar bombas de uranio, ha de cumplir 4 misiones a cual más arriesgada. STAINLESS STEEL es el juego más rápido que hayas visto nunca.

DISTRIBUIDOR EXCLUSIVO PARA ESPAÑA ERBE SOFTWARE  
C/ STA. ENGRACIA, 17 - 28010 MADRID, TFO. (91) 447 34 10  
DELEGACION BARCELONA, AVDA. MISTRAL, N.º 10. TFO. (93) 432 07 31

Steinar